

BI - Possibili scenari evolutivi

- Introduzione
- Legenda
- Esigenze mondo Ricerca
 - Una prima Analisi
 - Il primo obiettivo
 - Il secondo obiettivo
 - Scenario I, TUTTO SU DWH
 - Scenario II, ODS U-Gov+DWH
- Esigenze mondo Nuova Didattica (GDA)
- Note comuni con altri verticali
- Requisiti Funzionali (RF)
- Architettura Attuale
- PROPOSTA EVOLUTIVA
 - Architettura
 - Principali caratteristiche
 - PERFORMANCE
 - STORAGE
 - FONTI DATI
 - PROCESSO DI REPLICA e CARICAMENTO Lx
 - REPORTING e AUTORIZZAZIONI SUI DATI
 - VINCOLI
 - DELIVERY
 - SEMPLIFICAZIONE e AZIONI PREVENTIVE
 - GESTIONE TRANSITORIO
 - ASPETTI COMMERCIALI
- MINUTA Incontro TAD/Automazione
- MINUTA Incontro TAD/GDA
- Impatti legati alla sicurezza (GDPR)
- Evoluzioni DWH/KPI
 - Come SPRINT può leggere i dati dal DWTRASV
 - Come il DWTRASV può leggere i dati da SPRINT
- Esposizione oggetti ODS a query clienti
 - Possibili soluzioni
 - UNIBO
 - UNICAL
 - UNIPA
 - UNIMORE
 - Soluzioni valutate ma escluse
- Evoluzioni ODS/OS3
 - Situazione attuale
 - ODS IN GENERALE
 - SVILUPPO
 - CONFIGURAZIONE DI ATENEO
 - REPORT
 - PARTICOLARITA' OS3
 - Problemi attuali
 - Tamponi momentanei
 - DELIVERY
 - SVILUPPO
 - Proposte evolutive (pre incontro interno)
 - Proposta evolutiva (post incontro interno)
 - Presupposti:
 - Punti di attenzione
 - Soluzione proposta (nel breve)
 - Soluzione proposta (nel medio periodo)

Introduzione

Nel documento “.\\CVSWork\\Analisi Funzionale\\U-GOV PTH\\Analisi Strumenti OLAP 2017\\Analisi Strumenti OLAP.docx” si è discusso su come nel 2017 fossero utilizzate le piattaforma Microstrategy e Pentaho da parte degli atenei, dei principali pro e contro dei due e di quali siano le attuali architetture messe in campo da Cineca per erogare il servizio nonché un dimensionamento dell'architettura stessa in termini sia di utenti utilizzatori che di “ferro” utilizzato e di storage necessario.

In questa pagina wiki, invece, si vogliono analizzare quali possono essere eventuali scenari architetturali evolutivi futuri per adattarsi a quello che viene definito “l'affettamento dell'elefante” ovvero la riscrittura dei vari verticali ugov in moduli a se stanti interoperabili esclusivamente via webservices, ed anche, per poter espandere il DW/ODS con informazioni provenienti da fonti dati non oracle o, perlomeno, non ugov (es: mondo IRIS). Allo stesso tempo si vuole analizzare un'evoluzione che permetta di scaricare l'ecosistema ugov dal carico indotto dal reporting.

Per vagliare i possibili scenari si partirà da un'esigenza cogente, ovvero il modulo Didattica che è già partito con la propria riscrittura all'esterno di ugov (prodotto GDA) e con IRIS, esterno ad ugov, di cui si vuole incrociare la reportistica col mondo PJ.

Legenda

DM	Data Mart
PTH	Pentaho BI Platform
PDI	Pentaho Data Integrator (ETL Platform)
RDBMS	Database Oracle
TAD	Team CINECA che si occupa di Trasversale Analisi Dati
DW o DWH	Data Warehouse
ETL	Extract Transform Loading (processo di caricamento del DW)
PL /SQL	Linguaggio di scripting di Oracle
MSTR	Microstrategy BI Platform
ODS	Operational Data Store
OLTP	Istanza Oracle dedicata ad ospitare i dati Gestionali
OLAP	Istanza Oracle dedicata ad ospitare i dati dei DWH
UGOV o SIA	University Governance
BI	Business Intelligence
DBG	Database per la Reportistica Gestionale
DBO	Database per la Reportistica Operativa
DBB	Database per la Reportistica di Business Intelligence
Reportistica gestionale	La reportistica “funzionale all'uso del sistema gestionale” deve essere realizzata all'interno del sistema gestionale. (si tratta di reportistica che risponde alle domande “fammi vedere i dati che ho inserito”, “fammi estrarre l'elenco dei dati che ho inserito nel periodo ...” ecc...) e deve dare dati real time o near-real time senza necessariamente una profondità storica
Reportistica operativa	La reportistica “operativa” è un tipo di reportistica non necessariamente esposta e sviluppata all'interno del gestionale e con strumenti del gestionale; può avere profondità storica e un ritardo temporale sui dati di al massimo 1 giorno. Generalmente non prevede crosstab ma solo report formattati
Reportistica di analisi	La reportistica di analisi è quella a supporto della BI di ateneo e generalmente si appoggia su un datawarehouse e a report crosstab, grafici e cruscotti. Contiene la profondità storica dei dati e ha un tempo di refresh che può andare dal giornaliero al mensile

Esigenze mondo Ricerca

Gli obiettivi della ricerca sono sostanzialmente due:

- poter avere a disposizione report/cruscotti con indicatori di performance;
- permettere di incrociare i dati di U-Gov PJ (e CO) con i dati di IRIS AP con la creazione di reportistica da rendere fruibile all'utente finale (segretari di dipartimento e amministrativi, docenti ricercatori e loro delegati).

Una prima Analisi

Il primo obiettivo

pone la necessità di creare un DW Ricerca anche per i moduli IRIS AP e RM, tenendo presente che, ad oggi IRIS stesso ha già un proprio data mart sul modulo IR (necessario quindi definire come gestire questo aspetto anche in termini di politica interna di distribuzione dei ricavi); tutti e tre questi DW dovranno essere accessibile al DWTRASV se si vogliono realizzare cruscotti trasversali.

Come ipotesi minimale si potrebbe creare solo lo strato DB, senza la presentazione all'utente finale per l'analisi del dato, per quanto già gestito da IRIS (possibili problemi poi di duplicazione dei dati esposti, reporting IRIS vs dati dei KPI).

Si potrebbe direttamente utilizzare l'ODS di livello 1 già presente in IRIS come sorgente per la creazione di strati superiori per la reportistica DWH. Il database alla base di IRIS è Oracle e ci si immagina, come per UGOV ed ESSE3, di prevedere un accesso via dblink o con exp/imp ai dati dalla staging area del DWH.

Ad oggi non esistono WS in Iris che espongono tutti i dati degli ODS per cui ci si immagina che l'accesso diretto al db sia imprescindibile (visto anche che alcuni ods hanno una mole di dati elevata).

Il secondo obiettivo

pone la necessità di trovare un posto nell'attuale architettura in cui incrociare dati provenienti da due gestionali differenti che condividono via WS solo le anagrafiche; su questo ambiente si dovrà poter eseguire reportistica operativa richiamabile dal gestionale UGOV come avviene oggi per la reportistica di PJ; le possibili problematiche architetturali si possono riassumere in:

1. presenza delle istanze IRIS/UGOV/DW su istanze DB diverse su sistemi distribuiti e conseguente complessa gestione delle dipendenze e versionamento;
2. strato DB dell'ODS U-Gov con livelli >1, ad oggi gestito dal gruppo TAD, oneroso sia in termini di manutenzione che di esecuzione in quanto presenta degli ETL di discreta complessità (in particolare ODS L2 COAN-PJ).

Questi due punti aprono a diversi scenari e possibili conseguenze:

Scenario I, TUTTO SU DWH

Il DW attinge dati dalle due diverse fonti (UGOV e IRIS) e replica le tabelle/viste di interesse che vanno a formare la staging area per l'effettiva creazione del DW. In questa ottica DWH gestisce in autonomia il caricamento dalle fonti esterne indipendentemente dalla loro dislocazione e tipologia. Tempistiche di caricamento, gestione degli errori e qualità del dato vengono gestiti dal DWH. Questo comporta che i dati potranno avere presumibilmente un aggiornamento giornaliero (notturno).

In questo caso risulta necessario approfondire se i requisiti utente verrebbero soddisfatti in quanto:

1. deve essere verificata l'eventuale comparabilità del dato tra reportistica PJ (basata su ODS) e reportistica ottenibile dallo strato DWH: vi sarebbe, come minimo, un disallineamento temporale dei dati.
2. per raggiungere una completa equivalenza tra l'attuale strato ODS COAN L>1 e lo strato DWH si renderebbe necessaria una significativa estensione dell'attuale DW CO (che riporta anche dati di PJ) per recuperare i dati necessari agli incroci con IRIS/AP in modo da non dover creare un ulteriori DW con potenziali problemi di riconciliazione del dato.

Questo scenario pone anche il problema del tipo di piattaforma per le applicazioni utente del DWH e relativa accessibilità della reportistica erogata da parte dei verticali che debbano proporla ai propri utenti: ad es. autenticazione e profilazione.

Questo comporta una dipendenza funzionale reciproca tra verticali e DWH dove i primi necessitano del secondo per offrire reportistica operativa trasversale, il secondo deve recepire requisiti e supportare i report delivery richiesto dalla prime.

Scenario II, ODS U-Gov+DWH

I dati provenienti dai diversi verticali (Iris+U-Gov PJ-CO) devono transitare tutti su un unico strato (potenzialmente lo strato XM di U-Gov). Le ETL potrebbero essere di due tipi: integrazione lato DB o lato verticali (interoperabilità applicativa), ovvero i moduli che necessitano di dati provenienti da altri devono negoziare via servizi i contenuti che intendono condividere. Nel caso in cui i dati di AP debbano confluire nell'ODS COAN PJ L>1, dovrebbe essere PJ a recepirli e esporli come ODS L1. In caso di integrazione lato db diretta su schema XM (modello attuale per i moduli U-Gov), IRIS dovrebbe gestire l'ETL e il relativo "modulo ponte" U-Gov per poter esporre il dato (vedi già citati problemi di versionamento e compatibilità). Questo scenario, a fronte di una significativa complessità, resta però il più vicino alla soluzione che già oggi (almeno per la reportistica PJ) risponde meglio ai requisiti utente in termini di contenuti, freschezza del dato e flessibilità nella creazione di reportistica custom e del relativo delivery.

Esigenze mondo Nuova Didattica (GDA)

Riprogettazione del componente di reporting applicativo (ora ODS) alla luce della progettazione dello sviluppo della nuova didattica.

Occorre considerare diversi aspetti:

- Requisiti utente
- Tempi di Refresh
- Processo di sviluppo
- Organizzazione
- Strumento
- Modello dati

Si vuole avere uno strumento di Reporting applicativo. L'utente "reporter" deve poter costruire i propri report.

E' il gestionale che guida dal punto di vista di:

- Requisiti
- Dati
- User experience
- Refresh

Gli altri aspetti vanno subordinati a questo

Requisiti principali:

- R1. Il reporting applicativo deve funzionare su un modello dati dedicato, che non impatti sull'applicazione
- R2. Deve esistere un metadato descrittivo del modello di reporting del gestionale
- R3. Utente finale: usa il report; Reporter di Ateneo: usa lo strumento per definire i report; Cineca: definisce il metadato
- R3. Il gestionale deve poter essere pensato come formato da diversi moduli, ognuno con i propri dati (microservizi)
- R4. Nessuna assunzione sulla localizzazione dell'applicazione (cloud)
- R5. Refresh near real time

Ipotesi di lavoro: Usare Microstrategy?

Problemi da smarcare:

- P1: Integrazione, autenticazione, autorizzazione
- P2: Licenze
- P3: Processo ed organizzazione

Note comuni con altri verticali

La separazione dei verticali di U-Gov (ad esempio Didattica) sta già portando a ragionare su questi aspetti. La separazione del DB comporta ragionare su come dovranno/potranno essere costruiti gli ODS di livello superiore a L1 e quindi l'eventuale possibile necessità di poter accorpate tutto solo a livello DW. Ci si chiede quindi se sia possibile prevedere una soluzione architetturale, a tendere, che permetta di astrarre dalle morfologie dei diversi repository fonte (Oracle, Mongo DB, etc.) pur mantenendo la capacità di disegnare un modello concettuale trasversale omogeneo, con livelli di servizio equivalenti all'attuale ODS. In questo momento si tratta di capire se possa già essere l'attuale modello di DWH o si debba ragionare su soluzioni diverse.

Lato requisiti utente si segnala la forte necessità di mantenere uno strato comune (che ad oggi è l'ODS di livello 2 o superiore ed il relativo metadato) che consente una veloce produzione di reportistica personalizzata ed in "near-real time".

La produzione di questa reportistica sui progetti (che necessita dell'utilizzo di dati relativi a diversi processi) utilizzando direttamente viste di interscambio tra moduli verticali era stata abbandonata in quanto richiedeva competenze che, almeno al tempo (ma ci risulta ancora oggi), non erano gestibili dai singoli verticali. Inoltre nel tempo, ODS L>1 è stato in grado di recepire requisiti relativi a misure trasversali non gestibili a livello di singolo verticale (es: Proiezione utilizzo budget) che hanno poi avuto una ricaduta diretta sui dati esposti dai singoli verticali. Sarebbe necessaria quindi una riflessione sulle modalità di raccolta, gestione e implementazione di requisiti funzionali trasversali con finalità non esclusivamente di analisi del dato.

Requisiti Funzionali (RF)

Si elencano i requisiti raccolti

		Livello importanza requisito	NOTE	SOLUZIONE
RF = REQUISITI FUNZIONALI				
RF.1 Performance				
RF.1.1	La reportistica operativa deve poter essere utilizzata senza incidere sulle prestazioni del sistema gestionale.	OBBLIGATORIO	ovvero non si vuole più che i report vengano eseguiti puntando a oggetti del sistema sorgente	PERFORMANCE
RF.1.2	La reportistica operativa deve risultare performante anche all'aumentare della mole dati	DESIDERABILE	difficilmente ottenibile se si usano viste con logiche di estrazione complessa e se il sistema storage sottostante non è di tipo SSD	PERFORMANCE

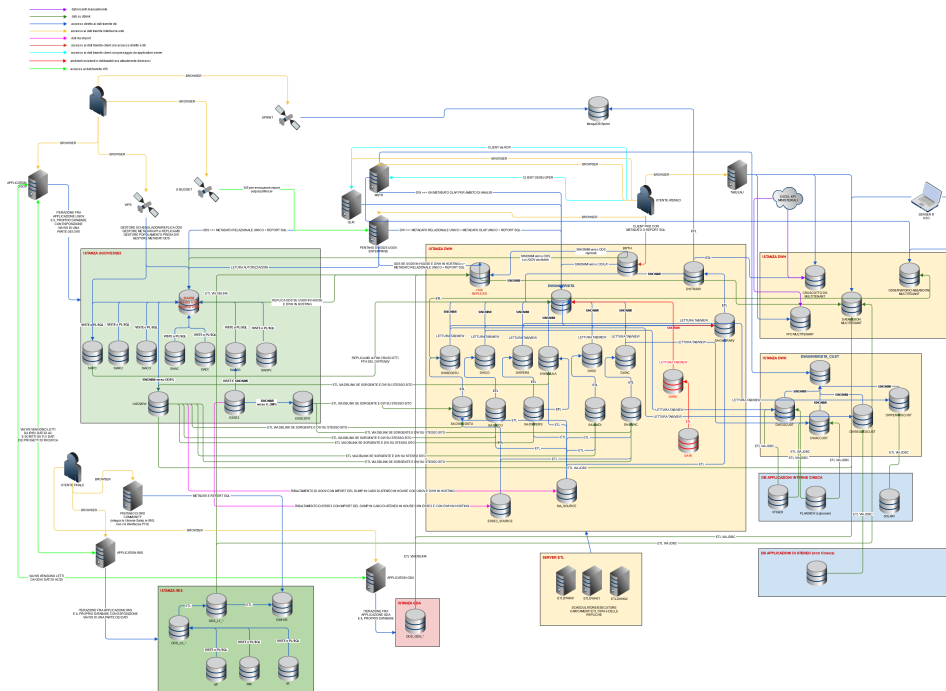
RF.2 Design dei report				
RF2.1	L'utente di Ateneo che ha determinate competenze e opportunamente formato, deve poter creare in autonomia nuova reportistica operativa	DESIDERABILE	gli atenei hanno già competenze in PTH e alcuni atenei anche su MSTR; già oggi sono in grado di creare reportistica in autonomia	già soddisfatto dall'attuale architettura
RF2.2	La consulenza CINECA deve poter creare in autonomia nuova reportistica operativa	OBBLIGATORIO	già avviene oggi	già soddisfatto dall'attuale architettura
RF2.3	Lo strumento di reportistica operativa deve consentire all'utente finale di realizzare reportistica in formato cross-tab	DESIDERABILE	Con PTH è necessario avere un metadato mondrian per poter realizzare reportistica crosstab; con Microstrategy la funzionalità è nativa. Su mondo ODS non esiste un metadato mondrian; era stato sviluppato ma poi dismesso; esiste invece sul mondo DW	la possibilità del CROSS TAB viene lasciata SOLAMENTE a I reporting analitico
RF2.4	Lo strumento di reportistica operativa deve consentire di costruire reportistica basata su istruzioni SQL	OPZIONALE	E' già così sia con PTH che con MSTR; dipende poi dai profili degli utilizzatori	già soddisfatto dall'attuale architettura
RF2.5	Lo strumento di reportistica operativa deve consentire la costruzione di report via web mediante funzioni di drag and drop di attributi, misure e filtri	OBBLIGATORIO	E' già così sia con PTH che con MSTR (tranne che per i report basati su sql o particolarmente complessi che prevedono l'utilizzo di un client)	già soddisfatto dall'attuale architettura
RF2.6	Lo strumento di reportistica deve consentire di applicare filtri complessi al report (AND e OR innestati)	OBBLIGATORIO	possibile solo con MSTR mentre con PTH è possibile solo impostare degli AND; eventuali condizioni complesse vanno precalcolate con dei flag su db e poi impostate su metriche filtrate	limite del prodotto PTH non superabile; necessario utilizzare MSTR
RF2.7	Lo strumento di reportistica deve consentire all'utente di Ateneo di realizzare reportistica operativa via web mediante l'utilizzo di un metadato	OBBLIGATORIO	Sia utilizzando PTH che MSTR è possibile creare un metadato su cui l'utente costruirà il proprio reporting (è l'attuale soluzione per tutti i DWH e ODS anche se per il mondo ODS è usato solo PTH)	già soddisfatto dall'attuale architettura
RF2.8	Lo strumento di reportistica deve offrire la possibilità di costruire report di prodotto realizzati da Cineca e messi a disposizione dell'Ateneo mediante sistema di deploy	OBBLIGATORIO	l'attuale ciclo di sviluppo funziona già così sia con PTH che con MSTR	già soddisfatto dall'attuale architettura
RF3 Integrazione applicativa				
RF3.1	SSO fra sistema gestionale e applicazione di reportistica operativa	OBBLIGATORIO	già così con PTH mentre è possibile con MSTR solo dalla release 11 (vedi mstr) che contiene una funzionalità per gestione dell'autenticazione con Shibboleth	a piano una migrazione a mstr 11 con sperimentazione della funzionalità; per PTH già fattibile
RF3.2	Possibilità di eseguire report operativi all'interno dell'applicazione gestionale	OBBLIGATORIO	la possibilità esiste già con PTH ma non è mai stata esplorata con MSTR di cui non se ne conoscono gli eventuali WS	PTH già lo permette; MSTR va sperimentato facendo una POC che metta insieme anche la possibilità di SSO
RF3.3	Possibilità di eseguire report operativi all'interno di applicazioni di terze parti (Ateneo)	OPZIONALE		
RF3.4	I dati esposti dallo strumento di reportistica operativa devono poter essere integrati anche con dati provenienti da altre fonti dati gestionali cineca (esse3, vecchi ods, verticali ugov.....)	OPZIONALE	ad oggi le uniche fonti dati utilizzabili da ODS sono quelle derivanti da UGOV e da ESSE3	A livello di Frontend solo MSTR permette (dalla versione 11) di avere data source di "terze parti" integrabili nelle dashboard insieme con i datasources di prodotto. A livello backend vedi FONTIDATI
RF4 Profilazione dati e autorizzazioni				
RF4.1	Lo strumento di reportistica operativa deve consentire di definire autorizzazioni sulla possibilità di visualizzare/eseguire report presenti all'interno del sistema	OBBLIGATORIO	già oggi possibile sia con PTH che con MSTR	funzionalità presente in tutti i tool di reportistica (poi da capire quando l'attività è automatica e quanto manuale)
RF4.2	Lo strumento di reportistica deve offrire la possibilità di definire filtri automatici sui dati visualizzati in base a regole impostate sull'utente che si collega	OBBLIGATORIO	già oggi possibile sia con PTH che con MSTR	
RF4.3	Lo strumento di reportistica deve offrire la possibilità di ereditare dal sistema gestionale i filtri da impostare sui report eseguiti dall'utente che si collega	DESIDERABILE	oggi è possibile solo con PTH (ereditando oltretutto le impostazioni dell'utente fatte su UGOV tramite una query sulla PTH madre)	Microstrategy non permette di ereditare in automatico da fonti date esterne eventuali valori da impostare nei security filter per un utente. Vanno impostati a mano
RF5 Freschezza del dato				
RF5.1	Il sistema di reportistica operativa deve mettere a disposizione i dati in tempo reale o con un ritardo massimo non superiore ad x ore	DESIDERABILE	Il tempo reale o il near-real time è difficilmente ottenibile e fuori dallo scope, per definizione, dei report operativi. Il ritardo di x ore invece è implementabile rispetto a vincoli di performance e architettura e bisogna che il cliente accetti questo ritardo	VINCOLI PERFORMANCE

RF5. 3	Il sistema di reportistica operativa deve consentire di effettuare un aggiornamento di dati su richiesta	DESIDERABILE	attenzione però alle performance; ad esempio, l'attuale sistema per UNIFI impiega 6 ore, non avrebbe molto senso la sua esecuzione on-demand	PROC-CAR
RF6 VARIE				
RF6. 1	Il sistema di reportistica deve essere lo stesso per tutti i possibili DBG seguendo il modello organizzativo di ODS (possibilità di attivare solo certi moduli di analisi)	DESIDERABILE	a oggi è già così con PTH, mentre MSTR serve solo la reportistica di analisi	il modello di reporting rimane come quello già in produzione
RF6. 2	Nel costruire la reportistica operativa deve essere possibile mettere in relazione metadati di aree gestionali differenti	DESIDERABILE	deve essere previsto a priori a livello di modellazione sia del metadato che della base dati; MSTR è più flessibile nel permettere di incrociare metadati, mentre PTH è più restrittivo e porta a dover modellare il tutto in modo più puntuale	FONTIDATI
RF6. 3	Le informazioni che dovranno poter essere interrogate dal sistema di reporting devono essere le medesime degli attuali ODS, ovvero il sistema deve essere compatibile con quello che già è in produzione in soluzione di continuità	OPZIONALE	la cosa è possibile se si sceglie di continuare con PTH come sistema di reporting ed espandere i BM esistenti mantenendo l'attuale struttura del metadato	PROC-CAR VINCOLI
RF6. 4	Lo strumento di reportistica dovrà permettere di ricevere aggiornamenti di release dei metadati e dei report senza dover riscrivere i filtri e gli accessi impostati prima del deploy	OBBLIGATORIO	a oggi è già così sia per MSTR che per PTH	rimane in essere quello già funzionante oggi
RF6. 5	Il sistema di reportistica dovrà permettere di creare report indipendenti dal "nome" degli oggetti usati e presenti nel metadato (dovrà lavorare per ID)	DESIDERABILE	MSTR lavora così "by design" invece PTH no per cui il cambio di un'etichetta può invalidare il metadato rispetto al reporting; è necessaria fare molta più attenzione in fase di mappatura	ci sono limiti nei software di reporting in essere e non ne è previsto al momento un cambio
RNF = REQUISITI NON FUNZIONALI				
xxx				
RNF . 1	La base dati del DBG non deve essere necessariamente una base dati Oracle	OBBLIGATORIO	tramite PDI è possibile accedere praticamente a qualsiasi tipologia di base dati	FONTIDATI
RNF . 2	La base dati del DBO e del DBD deve essere necessariamente una base dati Oracle	DESIDERABILE	tutta l'attuale architettura ODS/DW si basa sul fatto che la base dati sono Oracle; fu fatta anche un'analisi costi/benefici da ICTeam qualche anno fa sul spostare tutto su una nuova tecnologia e fu valutato che non ne valeva la pena	PROC-CAR ARCHITETTURA
RNF . 3	Il trasferimento dati fra DBG e DBO deve avvenire mediante servizi	DESIDERABILE	il problema potrebbe essere la mole di dati dato che spesso saranno trasferimenti massivi	PROC-CAR FONTIDATI ARCHITETTURA
RNF . 4	Il trasferimento dati fra DBG e DBO deve avvenire mediante data pump di dati da parte di DBG verso DBO	OPZIONALE	da valutare nel caso di situazioni distribuite; in realtà si vorrebbe studiare un'architettura che limitasse il passaggio dei dump senza dover passare tutta la base dati gestionale (sia UGOV che ESSE3); se si materializzano gli ODS in uno schema terzo l'ateneo potrebbe passare via dump solamente questo	PROC-CAR FONTIDATI ARCHITETTURA VINCOLI
RNF . 5	La gestione del trasferimento dati da DBG a DBO (schedulazione trasferimento dati, verifica esito trasferimento dati ecc...) deve essere monitorabile e debuggabile e non legata ad un'applicazione interna al verticale	DESIDERABILE	nova console di monitoraggio ????? l'attuale replica esistente scrive i log su una tabella e uno script notturno verifica la presenza di errori in questa tabella	PROC-CAR VINCOLI
RNF . 6	La DBO deve rappresentare la nuova fonte dati che va ad alimentare la base dati del DBB	DESIDERABILE	già è così; il DWH è alimentato solo dal mondo ODS. L'unica eccezione è il mondo segreteria studenti che legge direttamente da ESSE3. Si potrebbero trasformare tutte le attuali V_DM* in ODS_OS3 e far dipendere anche dwsegstu solo da ODS. Il fatto è questi ODS non verrebbero usati per il reporting ma solo per il caricamento del DWH col side-effect di aumentare molto lo storage richiesto per la replica degli ODS_S3*	ARCHITETTURA VINCOLI
RNF . 7	La gestione del trasferimento dati da DBG a DBO, e da DBO e DBD deve essere sincronizzato (es. se il trasferimento fra DBG e DBO va in errore, non deve essere caricato il DBB)	DESIDERABILE	questo è possibile se si fa seguire tutto il flusso di caricamento dati da un unico "gestore" e non cercando di sincronizzare flussi indipendenti	ARCHITETTURA VINCOLI PROC-CAR
RNF . 8	Le basi dati DBO e DBG NON devono risiedere su siti differenti (HOSTING e HOUSE)	DESIDERABILE	questo permetterebbe trasferimenti dati più veloci e con meno problemi legati ad ugovgw e a vincoli di apertura porte e visibilità db/db	ARCHITETTURA VINCOLI

RNF . 8	Le basi dati DBO e DBD devono risiedere solo in HOSTING	DESIDERABILE	questo permetterebbe di avere un reporting più veloce e un monitoraggio più puntuale e tempi di intervento più immediati. Questo implicherebbe anche di avere il sistema di reportistica in hosting che sgraverebbe molto le attività di configurazione/gestione in house	ARCHITETTURA VINCOLI
RNF . 10	Il DBO è un'unica istanza contenente tutti i dati di tutti gli atenei così come il DBG	OPZIONALE	dipende dall'architettura che si vuole mettere in piedi e se si vuole slegare il carico del reporting operativo da quello del reporting olap	NO, si sfrutta la stessa architettura di hosting del DWH in cui gli atenei sono splittati fra le varie istanze in un'ottica di carico
RNF .11	Il sistema di reportistica operativa deve poter gestire un numero elevato di utenze (>1000)	OBBLIGATORIO	al momento l'unico in grado di gestire questa mole di utenze è PTH in quanto i costi lato MSTR sono alti in mancanza di accordi OEM (o similari)	i sistemi di reportistica li gestiscono; argomento differente è se lo sforzo economico è sostenibile
RNF .12	Il DBO NON deve essere interrogato via SQL dagli utenti di ATENEI	DESIDERABILE	alcuni client (come report designer) necessitano di un accesso diretto al database; al massimo si può limitare l'accesso in sola lettura	LETTURAODS la lettura dovrà continuare
RNF .13	Il DBO deve esporre servizi per l'interrogazione di dati al di fuori della reportistica	DESIDERABILE	è necessario mettere in piedi un'interfaccia di gestione che esponga una serie di servizi che diano accesso ai dati	LETTURAODS non si svilupperanno WS ma si dà possibilità di accesso al db via sql
RNF .14	IL DBG non deve essere più direttamente interrogato via sql dagli utenti per l'accesso ai dati usati dal reporting	DESIDERABILE	l'utente via sql dovrà al massimo accedere alla replica di questi dati e non più lanciare query sul gestionale per avere i dati degli ODS	LETTURAODS
C REQUISITI ORGANIZZATIVI				
C. 1	La gestione del sistema di reportistica (aggiornamenti, verifica di anomalie, deploy ecc...) deve essere in carico all'area sistemi (IT&DP)	DESIDERABILE	si vuole mantenere l'attuale metodologia dove il team di riferimento viene ingaggiato in caso di problemi di delivery da parte di IT&DP	il delivery è sempre tramite Titanium come l'attuale architettura
C. 2	La gestione delle licenze del sistema di reportistica deve essere in carico a un gruppo "centralizzato" Cineca e non necessariamente al gruppo che lo sviluppa/utilizza	DESIDERABILE	separazione delle competenze; un conto è il licencing e le offerte /sconti che Cineca può ottenere e trattare col fornitore, un conto è chi lo strumento lo utilizza per i suoi prodotti che deve avere il contatto col supporto del prodotto	le licenze Pentaho ad oggi sono in mano a IT&DP, mentre quelle Microstrategy rimangono in carica al team TAD
C. 3	Il supporto interno sul sistema di reportistica deve essere erogato da un team esteso Cineca conscio del prodotto e di come è installato/configurato /gestito, non solo quindi dal team che ne sviluppa il prodotto su cui si basa la reportistica	DESIDERABILE	è un requisito che da tanto si vorrebbe poter perseguire in Cineca	non è prevista la creazione di un team di supporto dedicato
C. 4	Deve poter essere dichiarata in matrice di compatibilità la dipendenza fra DBG - DBO - DBD	OBBLIGATORIO	la matrice è l'unica a dare la garanzia del rispetto delle compatibilità in fase di aggiornamento di release	FONTIDATI
C.5	I report/metadati di prodotto dovranno essere parte (quindi essere deployati con) del DBO	OBBLIGATORIO	il versioning del metadato deve rispettare il versioning della struttura del database	il delivery è sempre tramite Titanium come l'attuale architettura e la reportistica /metadato rimane inclusa nel prodotto

Architettura Attuale

schema dell'architettura attuale



PROPOSTA EVOLUTIVA

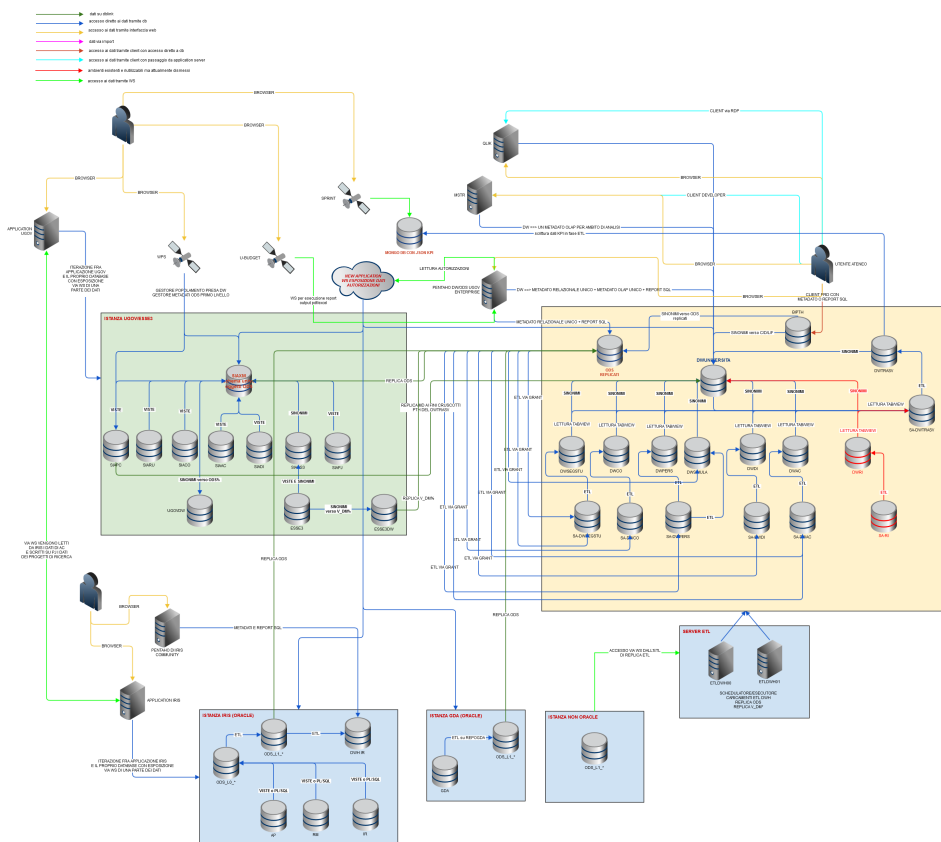
Si espande di seguito la proposta evolutiva che è in fase di approfondimento; se ne descrive l'architettura e le principali caratteristiche elencandone anche i pro e i contro e i punti ancora da approfondire

Issue di riferimento

ODS-2826
- Getting issue details...

STATUS

Architettura



Principali caratteristiche

Si propone una soluzione in cui si prevede:

PERFORMANCE

- unico schema accentratore di tutti i dati relativi agli ODS sull'istanza del dwh (uno schema UNlxx_ODS per ogni ateneo);
- il reporting insisterà tutto sul database del DWH quindi si toglie dall'istanza UGOV il carico dato dal reporting
- le performance del reporting migliorano sensibilmente in quanto le query punteranno sempre a tabelle e non più a viste del gestionale ("matriosche"); inoltre queste tabelle potranno essere indicizzate adhoc e potranno usufruire del parallel se risiederanno su Enterprise Edition
- si toglie dall'istanza UGOV il carico dato dai caricamenti ODS dei livelli >1 che si sposteranno sull'istanza DWH
- sdoppiamento del server ETL per reggere i carichi di memoria che aumenteranno dovendo gestire anche i job di replica; i job di replica non saranno quindi più gestiti/schedulati da WPS ma da uno schedule windows su server ETL preferibilmente come step iniziale di caricamento. Valutare un aumento di RAM del server

STORAGE

- Si libera spazio su storage ugov (a tendere) e, nell'immediato, se ne stoppa la crescita
- Aumento spazio occupato su storage SSD del DWH di hosting

FONTI DATI

- ai sorgenti dati, se oracle, si accede via dblink, ad altri tipi si accede con ETL PDI tramite server ETL; viste le moli dati in gioco la via principale sarà l'accesso diretto alla base dati (o via dblink). Con PDI è possibile accedere anche a web services via chiamate REST andando a popolare una tabella db in base al json ritornato; un'idea potrebbe essere quella di avere a disposizione due WS, uno che ritorni la lista degli ODS da replicare, e l'altro che ritorni la DDL oracle di questi ODS che vengono interrogati dal nuovo caricamento ODS. Non è infatti possibile da PDI creare job che prendono in input il nome di una tabella e la replicano....il job deve essere a conoscenza della ddl della tabella che deve popolare. NOTA: le nuove applicazioni in PaaS vengono realizzate in modo che ogni modifica venga messa subito in produzione su tutti gli atenei; quindi sono sempre tutti allineati alla stessa release dove quello che viene versionato è il microservizio esposto e non il "prodotto" nella sua completezza. Sarà da capire come integrare quindi questa modalità di gestione con la matrice di compatibilità di ODS.
- sarà da capire dove dovranno essere mantenute le informazioni riguardanti le connessioni ai database sorgenti (non ugov/esse3); si dovrà infatti trovare un automatismo affinché i dblink verso iris/gda vengano creati in automatico così come eventuali informazioni da inserire all'interno del jdbc.properties di pdi
- i primi livelli di ODS dovranno essere solamente delle viste o, se tabelle, popolate a carico del verticale
- possibilità di non avere più il dump di esse3 ed ugov in circolazione andando a replicare tutti i dati degli atenei anche in house nonchè tutti i dati esse3 non letti da ods (V_DM%); la replica delle V_DM è da valutare attentamente vista la loro mole

- i caricamenti DWH non andranno più via dblink ma leggeranno dalla replica locale degli ODS e delle V_DM%

PROCESSO DI REPLICA e CARICAMENTO Lx

- possibilità di discriminare da quale sorgente leggere
- ogni ods dovrà essere associato ad un particolare sorgente permettendo di discriminare quelli che sono, ad esempio, di UGOV o IRIS e quelli che invece sono "misti"
- dovrà sfruttare PDI per leggere gli ods dalle varie basi dati non oracle e scriverli in uno schema sull'istanza DWH di ateneo; sfrutterà un dblink per le basi dati oracle e potenzialmente potrà accedere a WS esposti dai verticali per l'accesso ai dati (invece che la connessione diretta al db)
- dovrà funzionare anche se un ateneo non ha il dwh; nel caso ci sia il processo di replica dovrà essere schedato prima del caricamento del DWAC e dei vari datamart in modo da essere pronti a fare in modo che i caricamenti dei dwh leggano dalla replica e non più dagli import (almeno per quando riguarda UGOV). NOTA: esiste uno step nel workflow pdi del dwh che si chiama IMPORT_PARALLELI che era stato pensato per integrare l'import di UGOV/ESSE3 in pdi ma mai utilizzato; capire se si può sfruttare questo step o se toglierlo
- nel caso in cui esista il dwh per l'ateneo, il processo di replica e i caricamenti dei datamart dovranno comunque poter essere eseguiti/schedati in modalità standalone oppure in modalità "integrata"
- dovrà sfruttare una funzionalità di switch o rename come per i data mart o i caricamenti degli ods attuali; il reporting non deve essere inficiato dalla replica (o avere un tempo minimo di down)
- dovrà replicare solo ods di prodotto presenti nella versione (quindi vanno tralasciati eventuali "cadaveri" di release precedenti) e quelli user defined (di cui non si saprà l'elenco)
- possibilità di replicare solo un determinato BM o solo uno specifico ODS; devono essere replicati solo oggetti ODS% ma non i %TMP, %FULL, % INCR
- possibilità di implementare dei meccanismi di dipendenza fra i vari ods in quanto la gestione dei caricamenti tramite pdi/sql sul dwh è più "libera" rispetto all'attuale gestione su UGOV; deve essere possibile, specificando di caricare/replicare un L2, di replicarne prima tutti gli L1 da cui dipende (e così via per L3, L4 e L5)
- sfruttare la dwh parametri per la gestione di eventuali parametri generali di flusso e unixx_tools per l'utilizzo di procedure/funzioni condivise (si potrà, stile segstu, creare tabelle di configurazione dello specifico ambiente dentro allo schema di replica come ad esempio la lista dei BM e relativi ODS, se replicare o no un certo ODS o BM, se replicare solo ODS di un certo livello, ecc..... questo per non sporcare la DWH_PARAMETRI con una lista molto ampia di ODS). Si avrà così maggiore libertà nella gestione/parametrizzazione dei caricamenti ODS dal momento che verranno gestiti in una modalità simile a come avviene oggi per il DWH (ad esempio, gestione del parallel per ateneo nei package dei livelli >1)
- dovrà poter personalizzare l'etl dei livelli successivi al 1° per i problemi, ad esempio, legati al parallel su certi atenei
- si dovrà poter schedare ogni ateneo al proprio orario e poter prevedere più caricamenti differenti per i vari atenei con una lista di ods caricati differenti
- schedulazioni dei caricamenti degli ods di livello >1 verrebbero gestite con schedule windows come avviene oggi per il DWH e non più con job oracle sull'istanza di UGOV
- gli ODS possono avere orari di replica e di caricamento differenti non più vincolati a determinate fasce orarie
- i nuovi package di caricamento degli ods di livello >2 dovranno essere riscritti in modo da non materializzare più in tabelle temporanee i livelli precedenti (es: ODS_L2_COAN_PJ)
- i nuovi package di caricamento degli ods di livello >2 dovranno essere riscritti in modo da prevedere la possibilità di un caricamento incrementale (si potrebbe replicare la logica implementata per il caricamento delle C/D del DWH)
- dovranno essere creati dinamicamente gli indici sugli ODS replicati allo stesso modo di come vengono creati oggi (vedi procedura del package di replica UNIXX_TOOLS.DWH_UTIL.crea_mv)
- alla fine della replica va abilitato il parallel sugli oggetti e computate le statistiche
- il tutto tracciato in log su database (si può sfruttare la già esistente LOG_REPLICA_ODS)
- si dovrà seguire una naming convention ad-hoc per la creazione dei package di caricamento degli ODS_USR% in modo che PDI non debba necessariamente essere a conoscenza del nome del package
- il monitoraggio delle repliche e dei caricamenti ODS_USR% o ODS_Lx dovrà essere monitorabile con neteye (in modo opzionale in modo che sia comunque compatibile con eventuali atenei in house) e avere log chiari e storicizzati e di facile consultazione
- dopo la replica del primo livello (unico che rimane sui gestionali) viene eseguito il caricamento di tutti i livelli successivi
- per ogni caricamento dovrà essere previsto uno step per il caricamento degli ODS_USR% che hanno un package di caricamento
- l'eventuale errore nella replica/caricamento di un ODS/BM non deve inficiare la replica/caricamento dei altri ODS/BM
- dovrà essere integrato con l'attuale workflow di caricamento del DWH in modo che prima si effettui la replica e poi i caricamenti (questi dovranno leggere dalla replica e non più dagli import)
- la password di SIAXM cambia/cambierà in fase di aggiornamento UGOV così come potenzialmente potranno cambiare quelle dei vari db a cui ci si deve collegare; questo farebbe andare in errore la replica degli ODS. E' giusto che sia così in modo da essere sicuri che non ci siano carichi /interferenze durante l'aggiornamento di UGOV. Allo stesso tempo però il GDPR ci impone di non avere cabalato nei dblink le credenziali applicative dei gestionali ma di utilizzare sempre utenze di sola lettura (sfruttare l'esistenza di UGOVDW/ESSE3DW che dovranno essere aggiornati nelle loro grant/sinonimi quindi non solo dal caricamento del DWAC ma anche dal processo di replica)
- se la sorgente è oracle allora la replica avverrà via dblink, se non è oracle si deve poter andare via JDBC tramite PDI. Inizialmente si si aspetta ci sia solo oracle ma conviene fin da subito creare "bussolotti" vuoti nel workflow e fare in modo che la cosa sia configurabile perlomeno inizialmente a livello di TipoSorgente (a tendere si dovrà arrivare al singolo BM e/o ODS, ad esempio nello scenario in cui un BM sia composto da un ODS acceduto via dblink e uno acceduto via REST)
- alla fine della replica di un ods o caricamento di un Lx dovrà essere data la grant di select all'oggetto creato alle utenze AU o BIPTH presneti (gestione grant puntuale)



Molte delle caratteristiche sopra elencate sono già state implementate nella logica della procedura crea_mv che sta alla base dell'attuale replica ODS pilotata da WPS; ATTENZIONE a utilizzare oggetti di unixx_tools che se usati frequentemente di giorno renderebbero difficile farne il deploy a causa dei lock. Preferire quindi una copia locale di oggetti di unixx_tools e poi esecuzione della copia locale



DA CAPIRE: su GIT va creata una struttura di cartelle utile al deploy e ai caricamenti; capire come gestire la parte "schema" ovvero la parte che contiene le ddl degli oggetti Lx (la si popola a mano ? si usa maven ? si cablano le ddl all'interno dei package di caricamento ?). creare un sottoaroma SCHEMA con dentro package e viste ? in fase di deploy come fare le diff stile maven o stile compare del deploy attuale ods ?



così come esiste la DWH_PARAMETRI per il caricamento del DWH valutare la creazione di una ODS_PARAMETRI e ODS_SCHEDULAZIONI per la replica/caricamento Lx e valutare l'adozione della gestione dell'id_set_parametri. Capire nel caso chi e come deve essere deployata (col dwh viene popolata dal processo di deploy del datamart con jenkins)



Attenzione alle schedulazioni multiple nella stessa giornata; le frequency si potrebbero lasciare a livello di giorno e se ci sono delle schedulazioni multiple in un giorno di alcuni ODS vanno gestiti con n schedule windows che lanciano n bat specifici dove si passa cabalto l'id set parametri da usare. Se parte la replica di tutto o un ods e c'è già un'altra replica in corso allora deve uscire diretto; se invece parte un caricamento con replica e poi dwh ma c'è un'altra replica in corso allora deve skippare direttamente alla parte di caricamento del dwh

REPORTING e AUTORIZZAZIONI SUI DATI

- l'idea era quella di avere la possibilità di accesso ai dati autorizzativi tramite WS tramite la creazione di una WebApp nuova che esponga una serie di servizi ad-hoc per la consultazione di questi dati e direttamente interrogabile da PTH così come avviene oggi con la xaction che punta al datasource SIAXM_AUTORIZZAZIONI e viene invocata alla login. Ad oggi, il sistema di recupero delle autorizzazioni funziona solamente tramite l'accesso alla PTH_MADRE di UGOV, quindi solamente con le anagrafiche censite su ugov e con i profili gestiti col relativo modulo di UGOV. Non c'è una soluzione per il recupero delle autorizzazioni dagli altri gestionali come IRIS o GDA. C'è un progetto Cineca di sviluppo di un'applicazione unica (Policy Manager) di gestione delle autorizzazioni di cui si parla da anni ma di cui è stata fatta ad oggi solamente la raccolta requisiti
- le autorizzazioni "automatiche" si hanno solamente con frontend pentaho (salvo eventuali approfondimenti di questa possibilità lato mstr); su mstr andranno rifatte a mano
- si mantiene PTH per il reporting operativo; per il reporting analitico si può utilizzare sia pentaho che microstrategy, mentre qlik rimane dedicato alle simulazioni

VINCOLI

- un ateneo in house per avere ods dovrà aggiungere un server windows per la gestione dell'etl così come dovrebbe fare se avesse il dwh (il db dedicato potrebbe non essere necessario...da valutare a seconda delle dimensioni dell'ateneo)
- aumento del carico giornaliero sulle istanze del DWH dato dal reporting PTH
- aumento del carico giornaliero sulle istanze del DWH dato dal caricamento degli ods di livello >1 e dalle repliche
- aumento occupazione RAM dovuto all'aumento delle JVM dei carichi sul server ETL del DWH
- i dati non saranno più in real-time dovendo accedere a delle repliche dei dati che avverranno 1 o due volte al giorno (da valutare la frequenza); questo non solo per il reporting (online o da client) ma anche per i dati acceduti da applicazioni esterne (es: PST)...da valutare alternative via dblink caso per caso
- le istanze del DWH non rientrano e non rientreranno nel disaster recovery e il recupero, nel caso di down, di un'istanza dwh è più "lungo" rispetto al gestionale. Il reporting, di conseguenza, non potrà avere gli stessi tempi di recupero che ha oggi (ne deriva che anche il reporting di UGOV, es. PJ, basato sui report PTH avrà tempi di ripristino lunghi nel caso di down dell'infrastruttura DWH)
- maggiore carico di rete sulle istanze del DWH indotto dalle repliche
- NON si potranno più definire report SQL che accedono direttamente alle tabelle dei verticali UGOV senza passare da ODS (regola che c'era già ma che non era un vincolo architetturale e quindi i report funzionavano comunque) (vedi nota per le eccezioni)
- la replica è prevista che sia solo in modalità full e non incrementale (almeno nel momento iniziale)
- eventuali ODS_USR% gestiti da WPS non potranno più accedere a ODS di livello >1 in quanto questi non si troveranno più su UGOV
- eventuali ODS_USR% gestiti da WPS potranno accedere solo a oggetti dei verticali che rimarranno in ugov (non si potranno ad esempio creare da WPS gli ODS_USR% per GDA o per IRIS)
- la replica degli ODS serve solo per "spostare" il puntamento dei dati usati dal reporting di pentaho; il data source delle autorizzazioni continuerà ad esistere e puntare a pentaho; PTH continuerà ad effettuare le query alla PTH_MADRE per recuperare le autorizzazioni degli utenti
- Il caricamento del DWH NON potrà più leggere dall'import di UGOV perché in UGOV non saranno più presenti i livelli successivi al 1°
- dal momento che alcuni ODS_USR% (ad esempio quelli di UBDG) leggono da ODS di livelli >1, necessariamente questi ODS devono essere caricati dopo tutti gli altri
- bisogna fare un'azione commerciale per portare gli atenei in hosting con il reporting e fare in modo che la rete fra ateneo e cineca sia sempre aperta; bisogna spingere tutti gli atenei a venire in hosting con il reporting e il relativo dwh in modo che tutte le repliche siano gestite e monitorate da Cineca; l'accesso ai db sorgenti dovrà avvenire da Cineca verso ateneo con aperture di rete specifiche (via UGOVGW) senza necessità di dover richiedere all'ateneo aperture temporanee di rete (vedi UNINA)
- i nuovi gestionali non ugov dovranno mettere a disposizione utenti di sola lettura per l'accesso agli oggetti ODS (si potrebbero sfruttare gli application user)



Esistono report SQL che accedono direttamente agli oggetti del verticale oltre che a oggetti ODS; ci sono tre possibili soluzioni:

- il report viene riscritto per utilizzare esclusivamente oggetti ODS andando eventualmente a creare nuovi ODS laddove mancanti
- il report, solo nel caso che non mischi oggetti del verticale con oggetti ODS di livelli >1, verrà eseguito configurando un DATASOURCE in Pentaho differente da quello utilizzato per il reporting ods (SIAXM). Questo permetterà, nel momento in cui si farà puntare il reporting alla replica, di mantenere questo nuovo datasource verso ugov
- se il report utilizza oggetti ods di livelli > 1 (in congiunzioni a quelli del verticale) o se ha delle peculiarità per cui è necessario che acceda a dati "freschi" ad ogni lancio, il workaround utilizzabile è quello di referenziare gli oggetti del report che puntano al verticale con il dblink che dovrà essere presente sullo schema unixx_ods. Questa casistica però deve essere LIMITATA a casi eccezionali

Alcuni di questi report utilizzano oggetti ODS_USR%; la problematica è la stessa. Alcuni ODS_USR% accedono direttamente a tabelle del gestionale; spostando gli ODS sulle istanze del DWH si potrebbe mantenere questa funzionalità utilizzando un dblink verso ugov e riscrivendo la query. Questa soluzione è già in uso per UNIPD per i report di UBDG in quanto in fase di budget l'ateneo ha necessità di reportistica con dati freschi e se i report si basassero sulla replica delle viste giornaliere l'ateneo non potrebbe correttamente lavorare

DELIVERY

- viene esteso il prodotto ODS esistente con nuove funzionalità senza creare nuovi PRODOTTI
- dovrà essere modificata la piattaforma applicativa Titanium per permettere il deploy di oggetti db sullo schema della replica e per deployare gli script etl sul server windows dei caricamenti del dwh
- si evolverà quindi solo l'ingegneria di delivery presente su Titanium (quindi basata sulla presenza di pth6); non si prenderà in considerazione la vecchia ingegneria hudson (quindi quella basata su pth4)
- i verticali NON UGOV dovranno integrarsi con Titanium in modo che al termine dei loro deploy, Titanium venga notificato della versione del prodotto installato sullo specifico ambiente; in questo modo il check di compatibilità di Titanium non dovrà essere modificato e sarà possibile specificare in matrice la dipendenza dei vari BM da IRIS, GDA o altri gestionali
- si dovrà gestire su dbconf la presenza di nuovi BM per ODS o l'associazione fra ODS non UGOV con BM già esistenti e che contengono oggetti di UGOV
- Titanium dovrà poter gestire connessioni a db differenti per il deploy, quello di UGOV per SIAXM e quello del DWH per UNLxx_ODS
- Titanium dovrà gestire il fatto che sia opzionale deployare la replica su un ateneo in quanto l'adozione sarà graduale
- il deploy dovrà funzionare sia che l'ateneo abbia già il dwh attivo che ancora da attivare (quindi non esisterebbe la struttura di cartelle sul server etl)

SEMPLIFICAZIONE e AZIONI PREVENTIVE

- rimarrebbero su UGOV solo gli ODS di 1° livello e sarebbero tutte viste; la gestione dei caricamenti degli ODS_S3 passerebbe o in pancia a ESSE3 o sui livelli superiori post replica (da capire se fattibile....vedi sezione ad-hoc su [os3](#))
- si potrà dismettere la funzionalità di REPLICATION ODS attualmente presente in WPS
- non si utilizzerà più WPS per la gestione dei metadati e monitoraggi degli ODS o, al massimo nel caso non si riescano a togliere i caricamenti dei primi livelli, l'interfaccia di gestione ODS esposta da WPS rimarrebbe per la parte di schedulazione degli ODS di 1° livello e per la creazione degli ODS_USR% che leggono solo da oggetti dei verticali (al più i primi livelli ODS)
- non verrà creata una nuova applicazione web per la gestione/monitoraggio della replica ODS e dei caricamenti dei livelli > 1; si dovrà agire come per il DWH andando a verificare i file di log, leggendo le mail e guardando le tabelle di log
- non verrà creata una nuova applicazione web per la creazione degli ODS_USR% sugli schema UNLxx (quelli che leggono oggetti di livello >1); visto che sono "pochi" e che gli atenei in tutti questi anni non ne hanno mai creati in autonomia, si è scelto di continuare a crearli solo manualmente direttamente sul db (team di prodotto/consulenza). Si dovrà definire una naming convention per i package di caricamento di questi ODS_USR% (se saranno tabelle) in modo che il flusso di caricamento PDI esegua il caricamento di tutti senza necessariamente doverne conoscere il nome
- si dovrà spostare sui verticali la schedulazione degli ODS di primo livello; a oggi gli oggetti sono:

Nome Package	Nome Tabella Target
ODS_CO92_003_ODS_CO_DIM_ANA_GE	ODS_CO_DIM_ANA_GERARCHIA
ODS_CO92_002_ODS_CO_PDC_COAN	ODS_CO_PDC_COAN_GERARCHIA
ODS_CO92_004_ODS_CO_PDC_COGE	ODS_CO_PDC_COGE_GERARCHIA
ODS_CO92_003_ODS_CO_UA_GER	ODS_CO_UA_GERARCHIA
ODS_CO92_006_UA_USER_CNTX	ODS_CO_UA_USER_CNTX
ODS_CO92_005_ODS_CO_UE_GER	ODS_CO_UE_GERARCHIA
ODS_PJ92_002_ODS_PJ_PROG_GER	ODS_PJ_PROGETTO_GERARCHIA
ODS_PJ92_003_ODS_PJ_UO_PROG	ODS_PJ_UO_PROGETTI
ODS_S3ODS92_064_ISCRITTIAPP	ODS_S3_ISCRITTI_APP
ODS_S3ODS92_094_LOGMASTER	ODS_S3_LOG_MASTER
ODS_S3ODS92_084_LTVRBTSTATO	ODS_S3_LOTTI_TR_STATO
ODS_S3ODS92_083_LOTTIVERB	ODS_S3_LOTTI_VERB
ODS_S3ODS92_020_QUESITIDOM	ODS_S3_P02_QUESITI_DOM
ODS_S3ODS92_018_QUESITIPAG	ODS_S3_P02_QUESITI_PAG
ODS_S3ODS92_019_QUESITIPARAG	ODS_S3_P02_QUESITI_PARAG
ODS_S3ODS92_021_QUESITIRISP	ODS_S3_P02_QUESITI_RISP
ODS_S3ODS92_046_P02_QUEST_COMP	ODS_S3_P02_QUEST_COMP
ODS_S3ODS92_047_P02QUESTCMPDID	ODS_S3_P02QUESTCOMP_DID
ODS_S3ODS92_095_P02QUESTCMPDOC	ODS_S3_P02QUESTCOMP_DOC
ODS_S3ODS92_038_P06AA	ODS_S3_P06_AA

ODS_S3ODS92_056_P06DIP	ODS_S3_P06_DIP
ODS_S3ODS92_028_P09_UD_PDSORD	ODS_S3_P09_UD_PDSORD
ODS_S3ODS92_061_P10APP	ODS_S3_P10_APP
ODS_S3ODS92_062_P10APPABILDOC	ODS_S3_P10_APP_ABIL_DOC
ODS_S3ODS92_063_P10APPLOG	ODS_S3_P10_APP_LOG
ODS_S3ODS92_031_UDLOGPDS_TCDOC	ODS_S3_UDLOGPDS_TCDOC
ODS_S3ODS92_096_V02GENQUESTDOC	ODS_S3_V02_GEN_QUESTION_DOC
ODS_S3ODS92_048_V02GENQUESTEST	ODS_S3_V02_GEN_QUESTION_EST

Il problema è che a oggi questi oggetti sono ancora un interregno dove la DDL della tabella viene rilasciata col prodotto ODS mentre il package di caricamento viene rilasciato col verticale. Soluzioni:

- Un'idea potrebbe essere quella che questi package vengono spostati sullo schema del verticale (non più su SIAXM) in modo che popolino, con una schedulazione del verticale stesso, una tabella locale al verticale e gestita dal verticale. Gli attuali ODS vengono trasformati in viste, sempre del verticale, che puntano a queste nuove tabelle.
In questo modo si avrebbe una completa scorporazione che tutto quello che è 1° livello è del verticale mentre i successivi livelli sono di ODS, con l'importante punto di attenzione in cui anche gli oggetti ODS di 1° livello sono parte integrante del prodotto ODS in quanto mappati sul metadato.
- Un'idea potrebbe essere quella di spostare nel job di replica anche tutti i package di caricamento degli L1 sopra elencati. Questo significa spostare però nel gruppo TAD anche le conoscenze del gestionale verticale o che comunque ogni eventuale modifica dovrà essere fatta a quattro mani e che il fatto che il package risieda lato TAD sia solo un tecnicismo per semplificare l'architettura. Quindi le viste di 1° livello rimarrebbero sul gestionale mentre package e tabelle di 1° livello passerebbero a TAD; in questo caso questi oggetti non verrebbero replicati e quindi in qualche modo il delivery di questa nuova architettura "ods" deve considerare che alcuni oggetti devono essere creati dal deploy e non solo replicati. Il vantaggio di questa seconda soluzione è che così facendo le schedulazioni degli ODS risiederebbero solo da un lato
- Un'altra idea è l'ibrido delle due precedenti riducendone la lista; infatti una cosa da considerare è che molte delle tabelle di 1° livello furono create per ragioni di performance. Vengono infatti popolate da una insert secca senza "trasformazioni" del dato, solo che la query è complessa e lasciarla come vista rallentava notevolmente il reporting. Essendo l'idea quella di materializzare su uno schema tutti gli ODS tabella, allora questi ODS di 1° livello potrebbero tornare a essere delle viste

Rispetto alla lista sopra queste sono:

ODS_CO92_003_ODS_CO_DIM_ANA_GE
ODS_CO92_006_UA_USER_CNTX
ODS_CO92_005_ODS_CO_UE_GER
ODS_CO92_003_ODS_CO_UA_GER
ODS_CO92_004_ODS_CO_PDC_COGE
ODS_CO92_002_ODS_CO_PDC_COAN
ODS_PJ92_003_ODS_PJ_UO_PROG
ODS_PJ92_002_ODS_PJ_PROG_GER
ODS_S3ODS92_038_P06AA
ODS_S3ODS92_019_QUESTIONIPARAG
ODS_S3ODS92_018_QUESTIONIPAG
ODS_S3ODS92_028_P09_UD_PDSORD
ODS_S3ODS92_031_UDLOGPDS_TCDOC
ODS_S3ODS92_020_QUESTIONIDOM
ODS_S3ODS92_021_QUESTIONIRISP
ODS_S3ODS92_056_P06DIP
ODS_S3ODS92_094_LOGMASTER

Rimangono invece complesse per cui è necessario mantenere il package:

ODS_S3ODS92_048_V02GENQUESTEST
ODS_S3ODS92_061_P10APP
ODS_S3ODS92_062_P10APPABILDOC
ODS_S3ODS92_063_P10APPLOG
ODS_S3ODS92_084_LTVRBTRSTATO
ODS_S3ODS92_064_ISCRITTIAPP
ODS_S3ODS92_083_LOTTIVERB
ODS_S3ODS92_095_P02QUESTCMPDOC
ODS_S3ODS92_096_V02GENQUESTDOC
ODS_S3ODS92_046_P02_QUESTION_COMP
ODS_S3ODS92_047_P02QUESTCMPDID

Per fare dei test di performance sugli oggetti di CO e PJ elencati sopra si possono eseguire su SIAXM queste query:



GESTIONE TRANSITORIO

Se si spostano i livelli > 1 da UGOV o si decide di convertire le tabelle di primo livello in viste è necessario che da una certa versione di ODS l'architettura /infrastruttura sia pronta ed in esercizio sull'ateneo. Ci dovrà essere un periodo transitorio in cui:

- FASE1: gli atenei che hanno già la replica attiva verranno riconfigurati disabilitando la replica pilotata da WPS e abilitando il deploy degli script di replica da Titanium; verrà configurata l'esecuzione della replica via schedulo lato server ETL
- FASE2: i livelli >1 saranno presenti sia su UGOV (con relativo caricamento) e su UNIX_ODS con caricamento rifatto; questo permetterà anche di fare la verifica di correttezza dei dati. In questa fase i livelli > 1 NON dovranno più essere replicati
- FASE3: tutti gli atenei che hanno ODS in hosting verranno configurati per deployare anche gli script di replica battezzando etldwh00 o 01 in base a dove attualmente si trova il dwh o in una logica di livellamento
- FASE4: si schedula la replica dei vari atenei in base alla capienza dello storage a disposizione sui db
- FASE5: si ripuntano man mano i vari datasource di PTH per puntare alla replica dopo una campagna informativa con gli atenei. In questa fase anche gli atenei in house col reporting dovranno passare alla replica eventualmente sulla stessa istanza di ugov, oppure passare in hosting col reporting
- FASE6: gli atenei che accedono con utenze PUEXT o AU agli ODS su UGOV devono essere "spostati" ad utilizzare gli stessi oggetti sulla replica
- FASE7: si trasformano gli ODS tabella di primo livello in ODS vista
- FASE8: i caricamenti del DWH devono puntare tutti come sorgente alla replica degli ODS (almeno per UGOV, mentre ESSE3 in questa fase può rimanere un import)
- FASE9: si tolgono gli ODS di livello > 1 dal deploy di ODS verso ugov lasciando solo la parte di replica

ASPETTI COMMERCIALI

- per gli atenei che hanno solo IRIS sarà necessario prevedere comunque una migrazione del loro reporting alla nuova architettura con replica ? o dovranno continuare a utilizzare il proprio Pentaho?
- come fare una campagna di comunicazione verso gli atenei ?
- come riuscire a portare tutti in hosting col reporting pentaho e la gestione della replica ?

MINUTA Incontro TAD/Automazione

Partecipanti: [Unknown User \(a.westcott@cineca.it\)](mailto:a.westcott@cineca.it) , [Unknown User \(m.castagna@cineca.it\)](mailto:m.castagna@cineca.it) , [Marco Brizi](#)

- il mondo ods sta andando verso un'architettura che prevede un replica degli oggetti ODS all'interno di uno schema delle istanze del DWH; la fonte dati degli ODS non sarà solo UGOV ma si necessita di attingere anche da IRIS, GDA e comunque potenzialmente da tutte le fonti dati delle nuove applicazioni che verranno implementate in Cineca in modalità microservizi
- non si vuole creare un nuovo prodotto ODS ma estendere l'architettura dell'attuale continuando a fare in modo che venga deployato da Titanium;
- essendo Titanium a fare il check di compatibilità di ODS verso gli altri prodotti della solution dichiarati in matrice, è necessario che ODS compili la matrice anche verso il prodotto GDA e IRIS e che Titanium sia a conoscenza delle versioni di questi due prodotti installati presso l'ateneo
- va trovato un modo per evolvere la piattaforma applicativa ODS per seguire questa evoluzione; in particolare:
 - servirebbe utilizzare uno degli step di aggiornamento della piattaforma applicativa per gestire il delivery degli script windows dell'etl della replica degli ODS (file presenti su git e da copiare sotto un certo percorso di un server windows da cui vengono eseguiti i caricamenti); va verificato:
 - se è possibile invocare "Carte" senza utilizzare il client di pentaho in modo che non sia necessario inserirlo nella piattaforma applicativa (dal momento che supera il GB di occupazione spazio) (vedi TOOLSPEC-1008)

- se è possibile far installare un server ssh sul server windows per eseguire questa copia da script shell (se lo step precedente dà esito negativo)
 - pensare ad una ulteriore strada
- va gestita nella piattaforma applicativa un insieme di variabili in più che contengano (almeno quelli immaginati al momento):
 - i parametri oracle di connessione allo schema UNLxx_ODS che conterrà la replica degli ODS e su cui andranno deployati i nuovi oggetti
 - i parametri di connessione al server windows contenente gli script di replica
- c'è un problema legato al recupero delle autorizzazioni in modo centralizzato dal momento che ogni prodotto che viene riscritto si riscrive a sua volta il modulo delle autorizzazioni sui dati; questo argomento dovrà essere riportato al gruppo di coordinamento
- per poter sviluppare le evoluzioni lato ods per leggere da IRIS e da GDA c'è bisogno di poter accedere ai dati di UNIBS (per GDA) e di UNIMI (per IRIS). Si è deciso per UNIBS di accedere a esse3/ugov presenti sul clone, mentre per UNIMI di accedere all'ambiente CINT UNIMI@qa02 già esistente e di cui Michele aggiungerà alla solution solo il prodotto ODS e Pentaho (vedi SDCINT-2356 e SDCINT-2357)

MINUTA Incontro TAD/GDA

Partecipanti: [Claudio Brizi](#) , [Unknown User \(a.westcott@cineca.it\)](#) , [Lorenzo Ficarra](#) , [Massimo Rimondi](#) , [Giovanni Introna](#) , [Unknown User \(n.santi@cineca.it\)](#)

- al momento è previsto che GDA sia solo su Oracle; è stato richiesto dagli atenei la possibilità di non avere oracle ma al momento si parte con oracle. Nell'ottica microservizi è comunque possibile avere GDA su, ad esempio, mysql ma tenere lo schema degli ods di gda su oracle
- sarà presente uno schema GDAODS (è il nome definitivo ?) in cui verranno creati gli oggetti ODS_GDA% che usciranno dall'analisi fra Rimondi/Brizi/Battipaglia (tolto dalla naming _L1 per dare 3 lettere in più utilizzabili per il nome "logico" dell'ods). Il popolamento di questi ODS è in carico a GDA che ne garantirà sempre la retrocompatibilità
- lo schema GDAODS verrà utilizzato sia per la creazione di oggetti utili al reporting ods, che per oggetti utilizzati dal reporting GDA che per l'esposizione di dati sul portale. Si cercherà di ricondurre le esigenze di reporting di GDA e del portale alle strutture dati che verranno richieste da ODS; se questo non sarà possibile verranno create delle strutture ad hoc che poi ODS non leggerà oppure verranno aggiunti campi alle strutture ODS che non varieranno comunque la granularità dei dati. A tendere i portali non dovranno accedere direttamente al db ma solo a servizi.
- le modalità/frequenze di caricamento degli ODS di GDA sullo schema GDAODS sono da concordare rispetto all'analisi fra Rimondi/Brizi/Battipaglia. GDA potrà sicuramente esporre dati sugli ODS che ha in pancia direttamente (es: id della persona) ma è comunque disponibile a valutare, in base all'analisi che emergerà, la possibilità di esporre anche dati non suoi ma recuperabili da altri servizi (es: nome/cognome della persona)
- il reporting ODS non avverrà sullo schema GDAODS; la ODS si implementerà una replica che permetta di consolidare in un unico punto tutte le basi dati ods dei vari applicativi. In questo schema accentratore, che starà sulle basi dati del DWH, verrà fatto puntare pentaho
- GDA vorrebbe ridurre al minimo il reporting effettuato direttamente dall'applicazione e "delegare" il più possibile il reporting allo strato ods
- al momento è previsto che GDA sia presente solamente in hosting; se ci saranno delle soluzioni ibride (host/house fra DWH e GDA) si dovrà valutare l'implementazione architetturale da mettere in piedi
- verrà creato un prodotto (GDA?, GDAODS?...da definire) nell'anagrafica prodotti e verrà versionato in modo che compaia nella pagina <https://wiki.u-gov.it/confluence/display/DELIVERY/Versioni> e in modo che sia utilizzabile dal prodotto ODS per la dichiarazione in matrice di compatibilità
- verrà implementato un meccanismo per cui all'atto del delivery di una nuova versione di GDAODS sui vari ambienti venga notificata la cosa a Titanium; così facendo Titanium saprà la configurazione dell'ateneo e potrà correttamente effettuare il check di compatibilità per il deploy di ODS
- esistono due ambienti di DEV di GDA (dati di UNIBS e UNITO) e un ambiente di preprod (pilota di UNIBS); l'ambiente con UNITO non è al momento utilizzabile per poter iniziare a sviluppare ODS, quindi si opta per l'accesso da parte di ODS via dblink all'ambiente di DEV di UNIBS. Questo ambiente di DEV al momento è collegato via WS all'UGOV di produzione di UNIBS; questo ambiente dovrà essere collegato/allineato all'ambiente di CINT di UNIBS che ODS concorderà col gruppo di Brizi in modo da poter iniziare gli sviluppi della replica lato ODS nonché gli sviluppi delle modifiche lato DW a fronte della doppia gestione DI/GDA che si dovrà implementare
- GDA ha un proprio strumento autorizzativo per la profilazione delle utenze. Al momento è prevista principalmente l'associazione fra un utente e il corso di studio (?); in ogni caso non è integrato col sistema delle autorizzazioni di UGOV e non presenta tutti gli elementi su cui è possibile effettuare "filtri" autorizzativi sugli attributi di analisi di Pentaho. Il problema è di ampio respiro (vale anche per IRIS) e dovrà essere portato sul tavolo di coordinamento di prodotto (tema "app per la gestione centralizzata delle autorizzazioni"). Nel frattempo GDA si è resa disponibile a fornire delle strutture dati o dei servizi che possano essere interrogate da pentaho per il recupero di determinate autorizzazioni. Il tema dovrà essere approfondito ma al momento è secondario visto che attualmente sugli ods di didattica non sono state implementate autorizzazioni sul metadato
- da verificare da parte del gruppo GDA dove sia possibile per il gruppo ODS recuperare le informazioni di accesso ai db/schema oracle di GDAODS
- il gruppo GDA verificherà con IT&DP la possibilità di creare utenti oracle di sola lettura verso GDAODS da utilizzare per l'accesso via dblink da parte della replica ODS
- negli ODS di GDA saranno presenti anche tutti i dati storici attualmente presenti sugli ODS di DI (quindi non solo l'offerta attiva)

Impatti legati alla sicurezza (GDPR)

I dati nella replica vengono scambiati su canali in VPN (ugovgw) o acceduti tramite HTTPS (nel caso di accesso via rest a servizi); i dati vengono memorizzati sugli stessi database su cui risiede il DWH su utenze specifiche non comunicate agli utenti finali. Se un ateneo ha necessità di accedere ai dati gli verrà dato a disposizione un'utenza personale a fronte dell'autorizzazione del suo dsi.

L'accesso al reporting e il sistema di autorizzazione sui dati dei report non cambia, quindi non si variano gli impatti sulla sicurezza rispetto all'esistente.

A tendere i caricamenti del dwh potranno leggere dalle repliche e non dai dump evitandone quindi la memorizzazione su ftp limitiamo così il rischio di perdita di dati per il fatto che i dump sono in chiaro su ftp

Evoluzioni DWH/KPI

Le richieste di ateneo sul mondo KPI portano ad una necessaria integrazione fra DWH e Sprint. Infatti su DWH sono presenti tutti i KPI che derivano dai vari datamart/ODS (più eventuali fatti esterni); su Sprint sono presenti kpi provenienti da altre fonti dati (es: MIUR). Su sprint si vogliono poter vedere i dati dei kpi del DWH per poter legare obiettivi anche a queste fonti, lato dwh si vuole potenzialmente integrare nel cruscotto di ateneo le informazioni provenienti dalle altre fonti extra ugov.

Come SPRINT può leggere i dati dal DWTRASV

Si sono vagliate varie possibilità:

1. possibilità da parte di Pentaho di esporre report la cui esecuzione/esportazione producesse un json
2. possibilità di creare una transformation/job PDI invocabile tramite Carte con una chiamata web senza utilizzare i client pan/kitchen di pentaho
3. sviluppo di un'applicazione/servizio dedicato per l'esposizione dei dati di ods/dwh
4. produzione di file json da parte del caricamento ETL del DWTRASV e loro successiva scrittura su ftp o su un db mongo
5. produzione di file json da parte del caricamento ETL del DWTRASV e a sprint tramite chiamata REST in modo che poi Sprint memorizzi i file in un filesystem locale alla PaaS

A fronte di varie valutazioni, sperimentazioni e verifiche si è optato per la soluzione 4 in cui la scrittura avverrà sul db mongo centralizzato di Cineca e su cui già Sprint si appoggia

Si cercherà di fare in modo che questa soluzione diventi standard anche per le altre sorgenti dati

Come il DWTRASV può leggere i dati da SPRINT

L'unica soluzione possibile è quella di effettuare delle chiamate a dei WS che esporrà SPRINT. Il caricamento ETL del DWTRASV, tramite PDI con delle chiamate rest, scaricherà da Sprint dei json con tutte le informazioni relative a kpi e metadati delle fonti dati non ugov e le integrerà nel caricamento (il dettaglio delle modalità lo si dovrà analizzare); a tendere anche le info di prodotto del dbconf finiranno su Sprint e le varie MD del dwuniversità, sorgenti del caricamento del dwtrasv, dovranno essere popolate da json di sprint e non più tramite la replica MD pilotata da WPS

Esposizione oggetti ODS a query clienti

frequentemente viene richiesto al gruppo TAD di poter dare accesso al database (in hosting) per poter far fare in autonomia query agli atenei sugli oggetti ODS. Questo serve per eseguire:

- estrazioni non servite dai report che sarebbero "macchinose" da esportare da report per poi essere segmentati in porzioni di dati e inviate ai vari destinatari
- utilizzo degli ODS come sorgente per DWH custom presenti in house
- utilizzo degli ODS come sorgente per applicazioni custom degli atenei

Da tenere in considerazione che:

- il gruppo TAD non gestisce servizi web in grado di esporre i dati degli ODS se non tramite report Pentaho
- non si può dare accesso diretto allo schema SIAXM agli atenei
- bisogna, per il GDPR, garantire l'accesso al set minimo di dati che deve vedere l'utente quindi non si può dare accesso, in linea generica, a tutto l'oggetto di prodotto
- il deploy ods degli oggetti tabella, nel caso in cui ci sia una modifica alla ddl della tabella, prevede il drop/create della tabella stessa e quindi si perdono eventuali grant

Possibili soluzioni

1. verranno creati le utenze in questo modo:
 - a. creazione personal user (PUEXT) sull'istanza di UGOV per ciascun utente di ateneo che deve accedere ai dati;
 - b. creazione di un application user (AU) sull'istanza di UGOV per ogni subset di dati omogeneo che devono poter essere interrogati dai PUEXT sopracreati;
 - c. si creano su SIAXM n ODS_USR che espongono i soli dati di prodotto che può vedere l'utente. In fase di creazione dell'application user si comunicano gli oggetti ODS_USR di SIAXM sui cui l'AU deve avere la grant. Si comunicano ai vari PUEXT (che si collegheranno proxando l'AU) le query che deve eseguire per interrogare i vari ODS_USR in modo che la esegua direttamente o si crei una vista sul suo schema personale (che però non è sotto backup)

NOTA: se l'accesso deve essere fatto da un'applicazione esterna e non direttamente da un utente, allora non verranno creati gli utenti PUEXT ma solamente l'utente AU (vedi ad esempio issue [SDAMADB-4202 - Getting issue details...](#) **STATUS**)

2. come la 1) solo che viene direttamente creato un ODS_USR che è già la query utile all'ateneo; l'ateneo quindi non dovrà fare altro che una "select * from SIAXM_UNIxx_PROD.ODS_USR%"
3. soluzione simile alla 1) ma effettuata sul db del DWH e non sul db di UGOV:
 - a. se l'ateneo deve accedere tramite utente UNIxx_BIPTH con client Report Designer allora si considera già questo schema l'AU e verranno creati i soli PUEXT che lo proxano. Il popolamento dello schema UNIxx_BIPTH è a carico di Cineca in quanto deve contenere un sinonimo per ogni oggetto ODS o DW in modo da far andare i report con metadato (ogni notte viene ricreato via script il contenuto dello schema)

- b. se l'ateneo non deve usare Report Designer o comunque l'utente deve accedere ad un sottoinsieme di ODS, viene impostata la replica degli ODS per l'ateneo richidente e viene inserita la grant sulla tabella di configurazione gestita dalla procedura di replica (M_GRANT). Questo utente verrà quindi usato come fosse SIAXM, quindi si rientra poi nel caso 1)
- 4. si passa all'ateneo l'export di ugov (o dwh se richiedono accesso ad un datamart) in modo che se lo importi su un db in house e acceda come e dove vuole
- 5. se devono accedere ai dati degli schema del DWH (i vari datamart)

Si può trovare documentazione al link [UAO - Istruzioni](#)

A seconda della richiesta o del momento "storico" in cui arriva si potrebbe configurare più conveniente una o l'altra soluzione: ad esempio:

UNIBO

- caricamento DWH custom: ha il dwh in house e gli ODS vengono utilizzati come sorgente per il caricamento mensile del DWPERs; in questo caso è meglio dare accesso all'intero set di ods direttamente con un export/import
- estrazione periodica di un report simile a quello di PJ con pentaho: qui la soluzione migliore è la 3) in quanto UNIBO presenta anche frequenti problemi di performance. Per tamponare l'esigenza è applicabile anche la 1) o a seguire la 2)

UNICAL

- caricamento DWH custom: stessa soluzione di export/import fatta per Bologna

UNIPA

- estrazione periodica di dati non servita da report PTH: la soluzione adottata è stata la 2) dando le grant, non solo agli ODS_USR, ma anche a viste di prodotto (vedi issue [SDAMADB-4202 - Getting issue details...](#) STATUS). All'ateneo è stato creato solamente un AU in quanto la persona che doveva accedere agli oggetti ods era una sola (sarebbe stato più corretto creare utenze PU)

UNIMORE

- accesso da applicazione esterna ai dati di ODS: questa è la richiesta più vecchia che ci è stata fatta e a suo tempo si optò per l'esposizione degli ODS su schema EAPP. In questo caso Brandolini mise lo script di creazione delle viste su EAPP in modo che il fixer le ricreasse sempre e diede le grant sugli oggetti di SIAXM. Non è chiaro se sta ancora funzionando la soluzione e andrebbe ricondotta ad una delle soluzioni individuate

NOTA BENE: a tendere andrebbero tutti riportati alla soluzione 3), sia che siano in hosting, sia che siano in housing (o nel caso dell'housing fornire il solo export degli ods e non di tutto ugov)

Soluzioni valutate ma escluse

- esposizione di oggetti ODS su SIAIE: lo schema nasce come frontiera contenente oggetti mantenuti dal prodotto. ODS non gestisce nel suo deploy eventuali fix su SIAIE e comunque gli oggetti da "pubblicare" sarebbero oggetti custom di ateneo e non di prodotto
- esposizione di oggetti ODS su EAPP: è una soluzione simile alle 2) ma con la differenza che la grant degli oggetti ODS su SIAXM verso EAPP deve essere garantita e gestita dal prodotto



NOTA BENE: deve essere chiaro all'utente che Cineca non fa manutenzione sugli oggetti ODS_USR o su query extra prodotto fornite. Se qualcosa smetterà di funzionare per via di modifiche apportate al prodotto, l'utente dovrà aprire ticket e in quella sede si valuterà l'impatto della sistemazione del problema



Se in fase di richiesta della creazione dell'application user si specificano gli oggetti di SIAXM che l'utente deve poter vedere, queste grant non verranno poi perse in caso di aggiornamento ODS in quanto tutte le notti gira uno script che li ridà in base alla richiesta che era stata fatta (esiste uno store che tiene traccia di tutte le grant che devono avere gli application user)



Esempio di comunicazione da dare all'utente per giustificare la creazione dell'application user:

per adempiere a quanto richiesto dalla normativa sul GDPR prima di procedere è necessaria la supervisione da parte di un autorizzatore che scriva di persona in questo ticket l'accordo a procedere; nella lista gestita dal vostro Direttore dei Sistemi Informativi vediamo elencati xxxx@xxxx yyy@yyy che ho provveduto a mettere in cc a questo ticket.

Non possiamo più creare account impersonali; creeremo un account personale per ciascuna persona che dovrà accedere.

Quindi, oltre all'autorizzazione delle persone sopra citate, è necessario fornire le seguenti informazioni per ogni utente:

Nome

Cognome

Data di Nascita
Luogo di Nascita
Codice Fiscale
Indirizzo mail

Si può specificare inoltre che nel momento in cui verrà censita l'anagrafica riceveranno una mail con le spiegazioni su come cambiare e mantenere aggiornata la password.

Una volta ottenuta l'autorizzazione dall'approvatore e i dati richiesti dell'utente, si deve aprire un ticket a SDCIS chiedendo che quelle persone vengano inseriti nel ramo LDAP PEOPLE di cineca.

Una volta che sistemi informativi chiude il ticket, va aperto un ticket a SDAMADB con la richiesta di creazione di un utente personale sul database dell'ateneo facendo riferimento al ticket CIS in cui sono state censite. Va specificato nel ticket anche gli indirizzi/range IP pubblici da cui si collegano dai pc dell'ateneo.

Evoluzioni ODS/OS3

Situazione attuale

ODS IN GENERALE

ODS rientra fra i prodotti della fabbrica dell'area TAD

ODS, dal punto di vista tecnico, è un prodotto composto da oggetti DB (viste, package, tabelle) e MW (metadati e report Pentaho) che risiedono sullo schema SIAXM di UGOV

Da un punto di vista logico è distinto in:

LIVELLI:

L0 (Livello 0) e Lx (Livelli da 1 a x); solo i livelli da 1 a x sono mappati da metadati di business e utilizzati per la costruzione di report senza scrittura di SQL.

- Il livello 0 è sviluppato dalle fabbriche verticali UGOV/ESSE3 [oggetti CO_ODS_%. Non esistono per ESSE3, ci sono delle viste...vedi spiegazione successiva di OS3]
- Il livello 1 è sviluppato a 4 mani fra le fabbriche UGOV/ESSE3 e la fabbrica ODS [oggetti ODS_CO_%]
- I livelli >1 sono sviluppati dal team TAD [oggetti ODS_Lx_abcd]

BUSINESS MODEL (BM):

Esistono uno o più BM per area funzionale che racchiudono al loro interno uno o più ODS che possono essere di vari livelli. Ad esempio il Business Model "STU - Indicatori ANS" è composto dagli ODS:

ODS_S3_P06_AA
ODS_S3_P06_CDS
ODS_S3_P07_NORM
ODS_S3_TIPI_CORSO
ODS_S3_P06_FAC_CDS
ODS_S3_P06_FAC
ODS_S3_P06_DIP_CDS
ODS_S3_P06_DIP
ODS_S3_P15_MON_EVENTI
ODS_S3_P15_MON_STAT
ODS_L2_S3_DIDATTICA_CDS

Lo stesso ODS può appartenere a più BM

Ad ogni BM viene associato un percorso fisico su cvs che conterrà eventuali report di prodotto relativi a quel BM.

Possono esistere BM che non prevedono report (quindi senza una cartella fisica associata) in quanto hanno solo oggetti db (come ad esempio ODS creati ad hoc solo ai fini del caricamento del DWH).

Possono esistere BM che non prevedono oggetti lato db in quanto sono usati solo come contenitori di reporting di prodotto legato ad ODS appartenenti a più di un BM

SVILUPPO

Durante la fase di sviluppo si specifica per ogni BM quali sono i verticali da cui si dipende, ma poi la matrice di compatibilità la si compila a livello di prodotto ODS e non a livello di BM.

Infatti si rilascia il prodotto ODS e non i singoli BM che quindi vengono sempre tutti rilasciati in blocco col prodotto ODS e ne ereditano lo stesso versioning.

Il progetto JIRA è unico per lo sviluppo ODS, così come è unico il metadato che mappa i vari oggetti ODS dei vari livelli (non è pensabile dividere i vari livelli ODS, quindi ad esempio non si può lasciare su UGOV il livello 0 e 1 e portare sul DWH i livelli successivi).

Il rilascio avviene mensilmente salvo urgenze per cui è possibile fare rilasci anticipati.

CONFIGURAZIONE DI ATENEO

Su ogni ateneo viene configurata la visibilità degli ODS ovvero quali BM devono essere deployati e se resi visibili al reporting (o perché ad esempio quell'ateneo non ha PTH come sistema di reporting ma usa MSTR oppure perché quel BM non prevede report associati).

Per ogni ateneo/ambiente/BM viene anche specificato il "gruppo di visibilità" ovvero quali saranno i/i gruppi LDAP che avranno accesso per l'esecuzione dei report contenuti nella cartella associata al BM.

Quindi a parità di release di ODS deployata su due atenei differenti ci potrebbero essere oggetti db ODS e report completamente differenti. Il motivo è che commercialmente non si vende "ODS" ma si vendono i BM della contabilità degli ODS o i BM dei questionari o delle risorse umane.

Inoltre, anche se un ateneo ha comprato i BM delle risorse umane, non è detto che tutti i BM di RU vengano resi visibili (per motivi legati al progetto)

REPORT

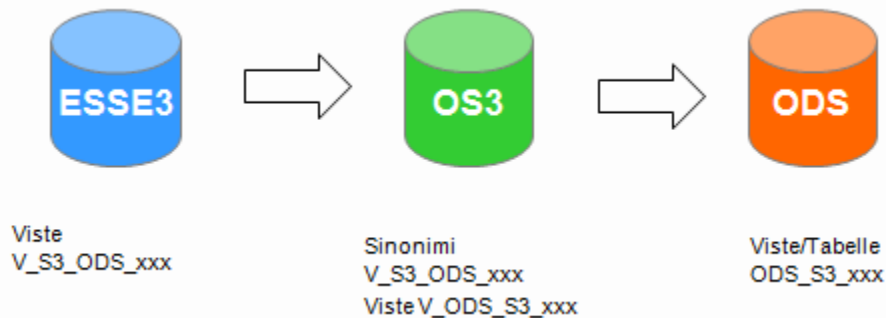
La fruizione dei report avviene da parte dei clienti:

- accedendo direttamente alla PUC
- tramite il client PRD sulle proprie PDL
- direttamente da UGOV sulle maschere del gestionale accedendo comunque alla puc, anche se in maniera trasparente per l'utente.

E' possibile in UGOV configurare dei diritti per i singoli utenti in modo da limitare la visibilità dei dati che verranno mostrati dai report eseguiti sugli ods dagli utenti stessi (si parla di profilazione sui dati).

PARTICOLARITA' OS3

Gli ODS basati sui dati di ESSE3, prevedono la seguente architettura di prodotti:



Ovvero:

- ESSE3 espone i dati mediante le viste V_S3_ODS_xxx che vengono rilasciate con le versioni del prodotto ESSE3;
- Il prodotto "tecnico" OS3 contiene: un sinonimo V_S3_ODS_xxx per ogni vista V_S3_ODS_xxx presente in ESSE3; una o più viste V_ODS_S3_xxx utili al caricamento degli ODS (solitamente V_ODS_S3_xxx = V_S3_ODS_xxx);*
- Il prodotto ODS contiene tabelle, package e viste ODS_S3_xxx che leggono i dati dalle viste V_ODS_S3_xxx;

Il prodotto OS3 è un prodotto "tecnico" nel senso che non ha un'interfaccia utente e non ha un proprio schema ma contiene solamente sinonimi e viste; ha un suo versionamento e viene aggiornato con gli aggiornamenti UGOV. Come per gli altri ODS, tutti gli oggetti sono scritti su SIAXM.

Quando UGOV non è collegato a ESSE3, i sinonimi di OS3 vengono fatti puntare a viste "FAKE" che si trovano su SIAXM. In questo modo è installabile OS3 anche in assenza di ESSE3 (utile ad esempio a poter deployare il BM PJ - Timesheet che prevede un oggetto di OS3 nel suo reporting..... CINECAERP, ad esempio, deve attivare questo BM e non ha ESSE3).

Il prodotto OS3 è composto da viste e tabelle; nel caso delle tabelle queste sono strutturate in modo conforme a tutte le tabelle ODS dei vari verticali e ciascuna tabella prevede un package di caricamento che ne gestisce il caricamento full e/o incrementale. Le viste e i package sono committate sul git di OS3 e installate col delivery di OS3, mentre le tabelle risiedono sul cvs/git di ODS e vengono deployate col delivery di ODS; i motivi di questa scelta risiedono principalmente in queste considerazioni storiche:

- la tabella doveva essere pilotata da TAD in quanto direttamente dipendente dal metadato e, quindi, dalla versione ODS che si va ad installare
- il package di caricamento e le viste contengono logiche di estrazione del dato sorgente le cui competenze non stanno in TAD ma nel verticale; TAD doveva concordare col verticale un tracciato record e il verticale provvederle al popolamento (anche se poi spesso la cosa veniva fatta a 4 mani).

La scelta relativa a se fare o no una vista è stato o:

- legata alle performance (ad esempio sugli ods legati agli appelli di esame o ai questionari) dal momento che il reporting punta direttamente a questi oggetti
- o legata alla complessità di estrazione del dato che non era direttamente ottenibile con una "semplice" select ma richiedeva una logica di estrazione complessa e il cui owner non poteva che essere il gruppo owner del dato (ovvero esse3)

A oggi i package di caricamento ODS (la relativa tabella di deduce dal nome) creati per ragioni di performance sono (sono un 1 a 1 con la relativa vista):

- ODS_S3ODS92_038_P06AA
- ODS_S3ODS92_019_QUESITIPARAG
- ODS_S3ODS92_018_QUESITIPAG
- ODS_S3ODS92_028_P09_UD_PDSORD
- ODS_S3ODS92_031_UDLOGPDS_TCDOC
- ODS_S3ODS92_020_QUESITIDOM
- ODS_S3ODS92_021_QUESITIRISP
- ODS_S3ODS92_056_P06DIP
- ODS_S3ODS92_094_LOGMASTER

Invece quelli creati con logiche complesse di estrazione sono:

- ODS_S3ODS92_048_V02GENQUESTEST
- ODS_S3ODS92_061_P10APP
- ODS_S3ODS92_062_P10APPABILDOC
- ODS_S3ODS92_063_P10APPLOG
- ODS_S3ODS92_084_LTVRBTTRSTATO
- ODS_S3ODS92_064_ISCRITTIAPP
- ODS_S3ODS92_083_LOTTIVERB
- ODS_S3ODS92_095_P02QUESTCMPDOC
- ODS_S3ODS92_096_V02GENQUESTDOC
- ODS_S3ODS92_046_P02_QUEST_COMP
- ODS_S3ODS92_047_P02QUESTCMPDID

Il motivo principale che spinse a suo tempo a predisporre il prodotto OS3 fu il disaccoppiamento fra i due prodotti ESSE3 e ODS. Si tratta infatti di un prodotto "light" che non ha al suo interno logica applicativa. Il disaccoppiamento doveva servire nei casi:

1. devono essere eliminate delle colonne di data base da ESSE3 e le colonne sono utilizzate da ODS;
2. deve essere rilasciata una versione di ODS che utilizza campi di ESSE3 aggiunti in versioni "troppo recenti" e non ancora installate o installabili sugli atenei.

Nel caso (1) è possibile eliminare le colonne da ESSE3 e modificare la vista V_ODS_S3_xxx che utilizza quel campo sostituendo la colonna con una colonna NULL as "nome campo". In questo modo l'ODS continuerà a funzionare e ci sarà tutto il tempo per adeguare gli ODS. Sarà quindi sufficiente rilasciare una versione del modulo OS3 che contiene la colonna NULL as "nome campo".

Nel caso (2) sarà possibile rilasciare una versione di ODS anche se su ESSE3 non sono ancora stati aggiunti i campi necessari all'ODS. In questo caso infatti sarà sufficiente rilasciare una versione di OS3 che contiene dei campi NULL as "nome campo".

Si tratta ovviamente in entrambi i casi di situazioni limite che raramente dovrebbero verificarsi. In ogni caso avere a disposizione il modulo OS3 può tornare molto utile soprattutto se si considera che il prodotto ODS è un prodotto con un unico versionamento che al suo interno include gli ODS di tutti gli ambiti di analisi.

Problemi attuali

L'architettura attuale richiede necessariamente un doppio vincolo di compatibilità fra ESSE3 e OS3 e fra OS3 e ODS; ciò significa che se ods necessita di esporre un nuovo campo dovrà attendere l'installazione delle versioni di OS3 con la struttura delle viste aggiornate che a sua volta deve attendere l'installazione di ESSE3 che contiene il nuovo campo nelle sue tabelle/viste di prodotto

Ciò significa che se in un aggiornamento UGOV è presente anche il modulo OS3 allora l'aggiornamento non potrà essere fatto fino a quando anche ESSE3 non avanzerà ad una versione compatibile; questo purtroppo non può sempre avvenire in quanto gli atenei NON vogliono legare gli aggiornamenti di ESSE3 a quelli di UGOV e viceversa, anzi spesso non lo permettono perchè le aree di ateneo e gli uffici sono differenti e difficilmente li si riesce ad accordare.

Un altro problema è legato al fatto che il ciclo di sviluppo/rilascio di OS3 è legato sia dal ciclo di sviluppo/rilascio di ODS che di ESSE3; capita che ods debba leggere una nuova colonna esposta da OS3 a sua volta esposta da una modifica fatta su ESSE3. Quello che avviene poi è che non si può rilasciare ODS perchè non si può rilasciare OS3 perchè ESSE3 verrà rilasciato fra tot giorni (....che è proprio quello che è capitato con il rilascio della 18.10 di ODS).

Quindi i problemi non ci sono solo in fase di delivery ma anche in fase di ciclo di produzione.

Tamponi momentanei

DELIVERY

Per tamponare è possibile far avanzare UGOV senza portare avanti il modulo OS3 e senza aggiornare ODS; questo però porta ad altri effetti indesiderati quali:

- si blocca l'avanzamento del reporting di CO/PJ senza che la cosa sia comprensibile per l'ateneo in quanto non capiscono il collegamento verso esse3

- IT&DP che si occupa del delivery deve gestire manualmente la cosa ad ogni aggiornamento ricordandosi poi di installare la nuova versione di OS3 e ODS quando verrà installata la nuova di ESSE3

Altro tampone è quello di, se possibile, applicare manualmente le modifiche su ESSE3 (spesso impatta solo delle viste) e poi procedere con l'aggiornamento di OS3/UGOV/ODS; anche qui però:

- IT&DP non può pianificare a "cuor leggero" gli aggiornamenti in quanto deve verificare prima se serve un intervento manuale
- qualcuno lato esse3 deve mettere mano alle produzioni facendo degli anticipi sui singoli ambienti

Altro tampone è quello di togliere la dipendenza fra OS3 e ESSE3 in modo che si riesca sempre ad aggiornare UGOV e, laddove si verificano degli errori di aggiornamento, si proceda manualmente a modificare gli oggetti OS3 aggiungendo delle colonne "null" dove necessario (sono poi questi i principali problemi, ovvero colonne aggiunte o tolte). Anche qui però:

- andranno in errore più aggiornamenti UGOV che richiederanno poi interventi manuali
- potenzialmente ambienti a parità di release OS3 avranno oggetti db differenti e quelli che si erano sistemati a mano vanno poi risistemati a mano quando viene poi aggiornato ESSE3
- un ateneo che ha aperto un change/fault riceve al rilascio della versione la comunicazione che la sua richiesta è stata risolta in una certa versione di ODS; a quel punto gli arriva poi la notifica che la versione di ODS è stata installata sul suo ambiente. Questo si collega e in un certo attributo vede tutti null invece che i valori che si aspetta. Dopo è complicato sia ricostruire la storia di quell'ambiente sia far capire all'ateneo perché non vede il dato

Altro tampone è quello di aggiornare UGOV e "staccare" ESSE3 facendo puntare le viste XM ai FAKE fino all'aggiornamento di ESSE3 alla versione di rottura; anche qui però:

- per IT&DP è macchinoso gestire l'aggiornamento;
- si romperebbe il reporting sugli ODS di ESSE3 finché ESSE3 non viene aggiornato (tutti report non avrebbero dati)
- le viste fake andrebbero rilinkate a mano al termine dell'aggiornamento di ESSE3

SVILUPPO

rilasciare comunque ODS e subito dopo deprecare la versione finché non viene rilasciato OS3

Proposte evolutive (pre incontro interno)

1. FICARRA: a suo tempo Giostra aveva gestito le "viste bridge" ossia, mi sembra, viste create dinamicamente che mantenevano la compatibilità con le versioni precedenti la rottura di ESSE3 creando come NULL i campi aggiuntivi di ODS. Magari approfondire il discorso in questa direzione
2. ALONZO/WESTCOTT: portare tutto OS3 (viste/tabelle/package) dentro a ESSE3 lasciando su SIAXM solo dei sinonimi ODS_S3* verso o i fake o gli oggetti di OS3 su ESSE3. Il modulo ponte è corretto che rimanga per i vantaggi scritti sopra, ma il suo delivery deve essere pilotato SOLO da ESSE3 interagendo con ugov per la v_xm_prodotti_installati, i sinonimi e i fake di XM. Questo permetterebbe anche di essere lineari rispetto a quanto si sta progettando come evoluzione del mondo ods dove gli ODS sono solo viste ed eventuali tabelle su cui si appoggiano stanno sul verticale; queste viste poi vengono replica sull'istanza del DW e utilizzate per il mappaggio del metadato diretto o il caricamento dei livelli successivi (la cui competenza logica/funzionale è corretto risieda in TAD)
3. FARINA: spostare OS3 dentro a ODS

NOTA:

fino a poco tempo fa l'aggiornamento di OS3 veniva effettuato sia da UGOV che da ESSE3. A giugno 2018, quando capitò un problema simile a quello di oggi sempre legato ad una rottura di compatibilità, ci incontrammo e si decise che per migliorare la situazione la cosa giusta da fare fosse fare in modo che OS3 venisse aggiornato solamente da UGOV e non più da ESSE3. Sinceramente non ricordo i dettagli delle motivazioni della scelta; ho solo recuperato questo scambio di mail in cui svariato delle cose scritte qui ce le eravamo già dette

da tenere in considerazione:

- tool e processo di delivery di ODS/OS3/ESSE3 (es: processo di delivery di OS3 gestito con dbfixer mentre quello di ODS no e una parte del mondo ODS è ancora su Hudson)
- competenze dei team in termini funzionali
-

Proposta evolutiva (post incontro interno)

partecipanti: [Libero Farina](#) , [Lorenzo Ficarra](#) , [Claudio Brizi](#) , [Stefano Buda](#) , [Unknown User \(a.westcott@cineca.it\)](#)

Presupposti:

- anche in ESSE3 è cominciata la suddivisione in moduli riscritti con le nuove metodologie a servizi
- nessuna garanzia che in futuro i vari moduli, o alcuni di essi, siano ancora su oracle
- non si vorrebbe più dare accesso alla base dati ma tutti i dati dovrebbero essere letti via servizi
- va smontato il castello di viste, sinonimi, grant incrociati per semplificare l'architettura e renderla più testabile sugli ambienti di CINT
- è necessario che del lavoro venga fatto sia lato ESSE3 che lato ODS

Punti di attenzione

- va gestito un transitorio prima di arrivare ad una soluzione definitiva
- lato esse3 si vuole la libertà di modificare la base dati garantendo però la retrocompatibilità di accesso al dato; quindi chi legge il dato non può andare direttamente sulla base dati ma deve sapere che servizio chiamare o che query eseguire dinamicamente
- a tendere, l'accesso via dblink al dato (quindi db to db) deve essere un'eccezione e non la regola. La discriminante sarà la mole di dati letta che si conferma essere un punto critico che può mettere in crisi la lettura del dato via servizi

Soluzione proposta (nel breve)

- le tabelle/package ODS% di OS3 presenti su SIAXM vengono "spostati" su ESSE3 e schedulati da ESSE3; perlomeno la logica è che il tracciato record che serve a ODS è quello delle tabelle già esistenti e il metodo per estrarre il dato è quello presente nei package. ESSE3 si riserva di scegliere il metodo migliore per dare a ODS lo stesso dato con lo stesso tracciato record, non necessariamente reimplementando in ESSE3 un package di caricamento
- le viste ODS% di OS3 presenti su SIAXM sono già una "select" di una vista lato ESSE3, quindi si suppone che lato ESSE3 non ci sia nulla da spostare/riscrivere
- viene mantenuto il modulo ponte OS3 che verrà deployato solo con ESSE3 e non più da UGOV
- il modulo OS3 gestirà l'interfaccia fra ESSE3 e ODS (così come oggi) e ODS continuerà a dichiarare la compatibilità solo verso OS3 (così come oggi)
- il modulo OS3 si occuperà di deployare su SIAXM un sinonimo ad un package di ESSE3 (nel caso esista ESSE3) oppure ad un package fake su SIAXM; questo package è sviluppato da ESSE3
- sia il package di ESSE3 che quello FAKE saranno fatti in modo per contenere al loro interno una table function per ogni vista/tabella ODS e dovranno essere invocati passando il numero di versione di tracciato record desiderato; in questo modo ESSE3 è libero di modificare la base dati esponendo una nuova versione della tablefunction che gestisca l'evoluzione della base dati. Allo stesso, nei limiti del possibile, dovrà garantire la retrocompatibilità della versione precedente della tablefunction rispetto alla modifica alla base dati, concordando col team ods un eventuale piano di dismissione di una versione della query. E' verosimile comunque che la maggior parte delle richieste evolutive arrivino proprio dal mondo ODS.
- il delivery di ODS dovrà scrivere su SIAXM tutti gli oggetti ODS che diventeranno tutte viste; queste viste dovranno essere fatte come select di una tablefunction presente all'interno del package deployato da OS3
- così come oggi, il delivery di ESSE3/OS3 e/o il relinking di UGOV con ESSE3 devono dare le grant di esecuzione a SIAXM a questo package e le grant di lettura agli oggetti di ESSE3

Soluzione proposta (nel medio periodo)

- sfruttando l'evoluzione architetturale di ODS e quanto già implementato nella soluzione precedente, lo script di replica degli ODS_S3 non dovrà più accedere a SIAXM ma direttamente a ESSE3; ovvero si deve vedere ESSE3 alla stregua di IRIS o GDA, ovvero fonte dati esterna ad UGOV
- lo script di replica invocherà sempre un package come nella soluzione precedente
- il caricamento del DWH legge dalle viste V_DM direttamente collegandosi alla base dati ESSE3; anche in questo caso la compatibilità andrà spostata da ESSE3 a OS3 e la lettura dovrà avvenire sempre utilizzando il package.
- questo package, sia per V_DM che ODS, però sarà differente rispetto a quello della soluzione precedente in quanto l'invocazione della package non ritornerà un recordset con già i dati, ma ritornerà direttamente la query che il processo di replica caricamento dovrà effettuare su ESSE3 per ottenere i dati:
 - questo permette di risolvere il problema che le tablefunction non funzionano se invocate via dblink
 - sarà da capire come gestire il dblink che dovrà essere scritto nella from delle tabelle restituita dalla query
- se verrà individuato un set di ODS/V_DM che restituiscono pochi record (da capire quanti sono "pochi record") si dovrà valutare la possibilità tramite PDI di invocare servizi per accedere a quei dati