

# **CONSIP S.p.A.**

## **APPENDICE 8 AL CAPITOLATO TECNICO**

### **Strumenti, standard e modalità d'uso per la gestione configurazione**

**Capitolato relativo all'affidamento dei servizi di sviluppo, manutenzione e gestione dei Sistemi di Vigilanza, Controllo e Pubblico Impiego del Sistema Informativo Integrato del Ministero dell'Economia e delle Finanze, Dipartimento della Ragioneria dello Stato.**

## SOMMARIO

<b>1</b>	<b>PREMESSA .....</b>	<b>3</b>
1.1	AMBIENTE MVS.....	3
1.2	AMBIENTE DIPARTIMENTALE/DISTRIBUITO .....	8

## 1 **Premessa**

Di seguito vengono descritti i prodotti che forniscono strumenti di gestione della configurazione.

Si sono considerati separatamente i due sistemi su cui insistono tali strumenti: MVS e dipartimentale/distribuito. Su tali sistemi appaiono diversi gli ambienti predisposti e i prodotti chiamati a fornire ausilio alla movimentazione di oggetti.

A prescindere dal sistema sul quale si opera e seguendo le specifiche tecniche dei diversi prodotti utilizzati, il fornitore deve effettuare le operazioni di trasferimento di tutte le tipologie di oggetti e controllarne il corretto esito.

### 1.1 **Ambiente MVS**

Il software utilizzato per le funzioni di configuration management in ambiente MVS è costituito dai prodotti riportati nella tabella che segue:

<b>Prodotto</b>	<b>Tipologia</b>	<b>Distribuito da</b>
DAJ	Creazione, modifica, documentazione e trasferimento JCL	Proprietario
FAA	Interfaccia ISPF per funzioni accessorie	Proprietario
Program Directory	Catalogo degli oggetti software	Proprietario
CCC/LCM	Configuration Management	ASG

**DAJ** (Documentazione Automatica Job) gestisce e controlla il rilascio in esercizio delle procedure JCL delle applicazioni batch (job, procedure e sysin); in particolare verifica la correttezza e completezza dei dati di documentazione operativa.

**FAA** (Funzioni Accessorie Applicative) è un dialogo ISPF che guida l'utente all'utilizzo di funzioni accessorie per il trasferimento di oggetti non movimentati da CCC/LCM, tra cui la gestione del dizionario dati DL/I, la gestione centralizzata dei messaggi diagnostici e alcune funzioni DB2.

**Program Directory** è basato su un database DB2 che viene automaticamente aggiornato al completamento del trasferimento di oggetti applicativi verso l'ambiente di esercizio.

**CCC/LCM** garantisce le funzioni specifiche di configuration management.

Consip si riserva la possibilità di variare tali prodotti, in funzione di attività di dismissione, peraltro già in atto, e di fornire strumenti e/o modalità alternative per gestire i rilasci, i trasferimenti e l'archiviazione delle informazioni.

All'interno del database di CCC/LCM sono memorizzati tutti gli oggetti software appartenenti alle seguenti categorie:

<b>Tipo CCC/LCM</b>	<b>Descrizione</b>
SOURCE	Programmi sorgente
COPY	COPY Fa*
DCLGEN	Dichiarative DB2 - Ta* Va*
INCLUDE	Include DB2 non dichiarative Da*
DGIPNL	Strutture sorgenti SDF (panel)
DGIGRP	Strutture sorgenti SDF (panel group)
MAPDSECT	Mappe logiche (xxxxMXX)
MAPSET	Mappe fisiche (xxxxMST)
PSBSRC	Sorgenti PSB

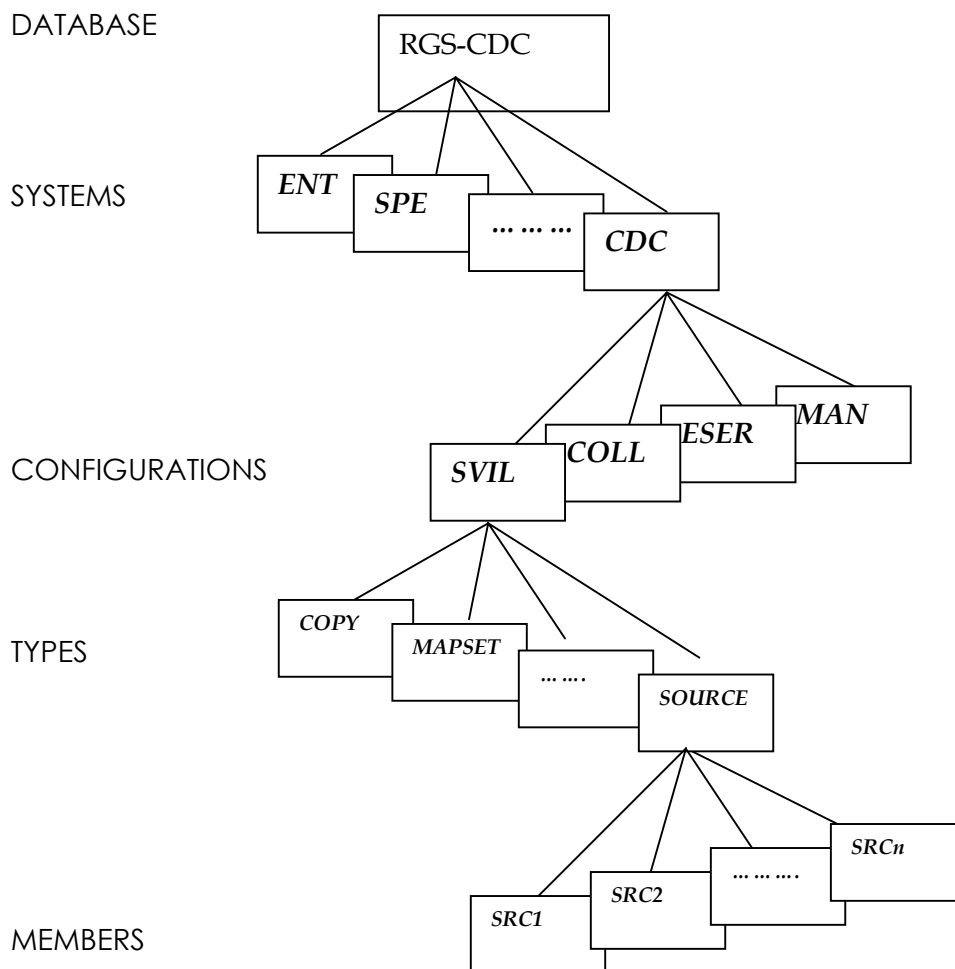
Nella tabella che segue sono riportate le possibili combinazioni di linguaggi/sottosistemi gestiti:

<b>Tipo</b>	<b>Cobol</b>	<b>Assembler</b>	<b>Metacobol</b>	<b>YM1001</b>	<b>Rcopy</b>	<b>CICS</b>	<b>SGC</b>	<b>DB2</b>	<b>DLI</b>	<b>MQ Series</b>
A	X		X		X			X		
B	X		X	X	X	X	X	X		
E	X				X	X		X		
F	X		X		X	X	X	X	X	
G	X		X		X			X	X	
K	X									
J	X		X	X	X	X	X	X		
L	X							X		
M	X					X		X		
Q	X					X		X		X
1	X		X		X					
2	X		X		X				X	
3	X		X	X	X	X	X			
4	X		X		X	X	X		X	
6	X				X	X		X		
7		X			X					
8		X			X	X				

Si fa notare che alcuni tipi sono ammessi solamente per specifiche aree applicative. Sarà cura di Consip comunicare le tipologie da utilizzare a seconda dei casi.

Consip si riserva la possibilità di apportare variazioni a tali possibili combinazioni di linguaggi/ sottosistemi.

La struttura logica del database di CCC/LCM prevede la seguente organizzazione gerarchica:



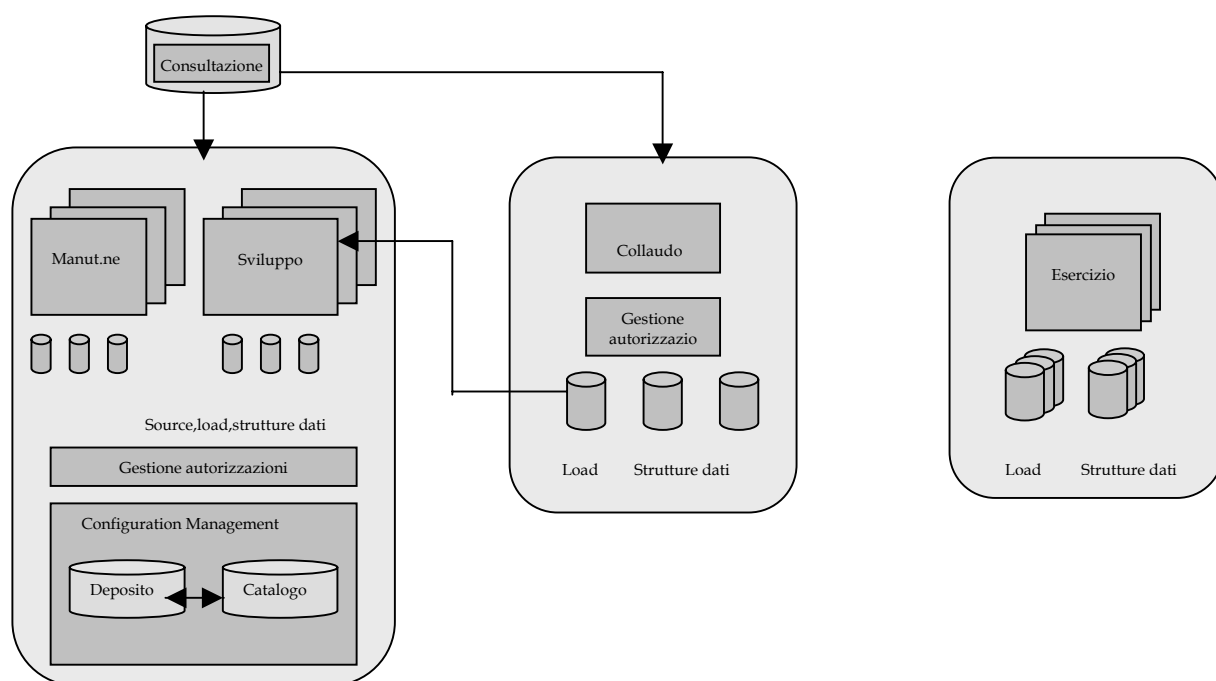
I **systems** corrispondono alle aree operative riportate nell'appendice 2.

Le **configurations** corrispondono agli ambienti di sviluppo, manutenzione, collaudo ed esercizio.

I **types** corrispondono alle tipologie di oggetti software (source, copy, mappe, ecc.).

I **members** corrispondono ai singoli oggetti software.

Nella figura seguente si riporta uno schema degli ambienti di sviluppo, collaudo, manutenzione, ed esercizio disponibili su sistema centrale MVS.



*Figura 1: Organizzazione degli ambienti di programmazione su piattaforma centrale*

L'attività di sviluppo e manutenzione avviene su elaboratori e ambienti messi a disposizione da Consip.

Il configuration management CCC/LCM garantisce l'isolamento tra gli ambienti e tra i fornitori, il governo e la gestione del ciclo di vita delle applicazioni. In particolare, esso sovrintende alla concorrenza nell'uso di oggetti (versioning, lock, ecc.) attraverso un suo "deposito", regola la distribuzione degli oggetti tra gli ambienti e tiene aggiornate le informazioni statistiche (dimensioni, date di movimentazione, ecc.), relative agli oggetti software, attraverso un suo catalogo.

Le attività svolte mediante le funzioni del prodotto sono la creazione degli oggetti, la compilazione dei programmi ed il trasferimento dei pacchetti software tra gli ambienti; le operazioni di editing sono invece esterne al prodotto ed avvengono su comuni librerie partitioned; il passaggio degli oggetti tra il database del prodotto e le librerie di editing avviene mediante funzioni di import/export.

La struttura del ciclo di vita del software prevede che i pacchetti realizzati in ambiente di sviluppo vengano trasferiti in collaudo e al termine del collaudo stesso vengano infine trasferiti in esercizio; i pacchetti realizzati in ambiente di manutenzione, invece, dato il carattere di urgenza che tali attività generalmente hanno, vengono trasferiti direttamente in esercizio.

Di seguito si descrivono i processi di movimentazione degli oggetti tra gli ambienti configurati dal CCC/LCM.

Per il passaggio da sviluppo a collaudo il fornitore dovrà preparare opportuni package contenenti l'insieme di job, procedure, sysin ed eventuale software specifico per predisporre l'ambiente di collaudo (definizione/modifica tabelle, caricamento dati, ecc) e il

software applicativo. La migrazione verso collaudo del package contenente gli oggetti da sottoporre a collaudo dovrà essere successiva alla predisposizione dell'ambiente.

Consip comunicherà al fornitore la nomenclatura e gli eventuali parametri da utilizzare.

Consip darà autorizzazione alla fase di trasferimento verso collaudo.

Dal momento del passaggio in collaudo gli oggetti non saranno più modificabili in ambiente di Sviluppo.

Nel caso di malfunzionamenti riscontrati durante il collaudo, le parti da sottoporre a correzione vengono regredite in ambiente di Sviluppo tramite apposite procedure per le necessarie correzioni. Tale attività di passaggio da collaudo a sviluppo può essere richiesta al fornitore.

Il passaggio da collaudo a esercizio è effettuato da Consip, utilizzando le procedure predisposte allo scopo dal fornitore.

Contestualmente vengono aggiornate le librerie di Consultazione e allineato l'ambiente alternato di manutenzione correttiva.

Gli oggetti sugli ambienti target (da sviluppo a collaudo e da collaudo a esercizio) sono disponibili, di norma, in circa 2 giorni.

Per la correzione delle malfunzioni, il fornitore dovrà trasferire gli oggetti da esercizio a manutenzione.

In ambiente di manutenzione gli interventi correttivi non comportano in alcun caso modifica della struttura della base dati. In caso di interventi che richiedano la modifica della base dati dovrà essere utilizzato l'ambiente di sviluppo.

La base dati dell'ambiente di correzione è perfettamente allineata a quella di esercizio e aggiornata conseguentemente ad ogni passaggio da collaudo verso esercizio. Il popolamento di tale base dati è periodica a cura Consip.

Nei casi in cui gli oggetti da correggere siano contemporaneamente in fase di modifica per un intervento di sviluppo, fermo restando il controllo del configuration management che segnala tale situazione, i conflitti e la successiva integrazione del software, devono essere risolti puntualmente con accordi specifici.

Al termine dell'attività di manutenzione, le applicazioni vengono promosse dal fornitore da correzione a esercizio tramite apposite procedure di trasferimento. Tale attività avviene, di norma, senza necessità di autorizzazione da parte Consip.

Contestualmente al passaggio in esercizio vengono aggiornate le librerie di Consultazione e allineati gli ambienti alternati di sviluppo e collaudo.

In tutte le situazioni descritte le attività e i prodotti (job, ddl, ecc) richieste al fornitore per il trasferimento degli oggetti tra gli ambienti (sviluppo, collaudo, manutenzione, esercizio) potranno essere ulteriormente specificate in funzione delle caratteristiche dei singoli progetti.

## 1.2 Ambiente Dipartimentale/Distribuito

L'ambiente di Configuration Management relativo agli ambienti dipartimentali sia UNIX sia Windows si basa sul prodotto ALLFUSION HARVEST CHANGE MANAGER.

E' in corso attualmente la migrazione dei progetti in ambiente Harvest.

Le indicazioni che seguono, pertanto, devono essere considerate quali indicazioni generali e comunque passibili di modifiche.

Tutte le librerie del progetto saranno definite al prodotto di configurazione, comprese le librerie di sviluppo.

Attualmente il server su cui risiede il prodotto di CM è un sistema SUN Solaris 2.6.

Gli oggetti posti in configurazione sono i componenti dell'applicazione, quali: form, report, programmi eseguibili, interpretati, sorgenti, maschere applicative, script utilizzati per la modifica della base dati, ecc.

Non è previsto il supporto di gestione configurazione con Harvest per i seguenti prodotti:

- Power Center
- OFA

Consip si riserva di sottomettere alla gestione configurazione anche tali prodotti, utilizzando, eventualmente, strumenti diversi da Harvest.

Il flusso operativo è il seguente:

- Due linee produttive del sw operanti in modo indipendente, eventualmente con contributo di fornitori diversi, una per lo sviluppo e l'altra per la manutenzione correttiva.
- Il fornitore che esegue lo sviluppo di nuovi obiettivi o di manutenzione evolutiva (SVIL/MEV) opera autonomamente rispetto alla struttura Consip.
- Il fornitore di manutenzione correttiva (MAC) opera presso postazioni definite da Consip.
- Consip è responsabile dell'allineamento delle attività tra le due linee.
- L'integrazione tra le diverse versioni della stessa sorgente, quella di sviluppo e quella di manutenzione correttiva, è svolta completamente manualmente e non tramite i processi Harvest (in modalità "out of the box").
- Per ogni attività di MAC che impatti su oggetti trattati da attività di SVIL/MEV, viene inviata la documentazione relativa al gruppo di sviluppo, preavvisato, perché provveda all'integrazione.

I gruppi di lavoro sono indipendenti e non utilizzano processi Harvest di concorrenza, pur gestendo tale evento in modo logico-organizzativo. L'integrazione tra sorgenti è supportata da comunicazioni automatizzate ed è eseguita esternamente al prodotto.

Il modello di riferimento per Consip, è un ciclo di vita caratterizzato dalla lavorazione degli oggetti software in modalità lineare, dove potrebbe essere realizzata anche una concorrenza virtuale.





1. "SVI-MEV"
2. "TEST\_SVI"
3. "MAC"
4. "TEST\_MAC"
5. "COLL"
6. "PROD"
7. "Attività Annullate"

E' possibile vedere il percorso tra tali stati in relazione a due diversi flussi operativi:

- Sviluppo di un nuovo progetto o di Manutenzione evolutiva.
- Manutenzione correttiva.

Ogni stato definito nel ciclo di vita ha associato una vista logica propria, ciò per facilitare eventuali ricerche sul repository. Fanno eccezione gli stati di SVI-MEV, MAC e PROD, che condividono la stessa vista, questo per poter gestire la manutenzione evolutiva e correttiva di ciò che è presente in produzione dagli stati di sviluppo e manutenzione.

STATO	VIEW
SVI-MEV	Produzione
TEST_SVI	Test-Mev
MAC	Produzione
TEST_MAC	Test-Mac
COLL	Collaudo
PROD	Produzione
Attività Annullate	Attività-Annullate

Harvest gestisce il software in un ciclo logico separato dalle reali macchine su cui le applicazioni saranno eseguite. Solo in alcuni casi saranno gestiti gli aggiornamenti sulle macchine reali, innescati a seguito di specifici processi utente (promozioni) ed operanti tramite scripts shell.

Promozione tra stati logici Harvest del ciclo di vita		Ambiente operativo fisico aggiornato in conseguenza
DA	A	
SVI-MEV	TEST_SVI	Collaudo
TEST_SVI	COLL	
MAC	TEST_MAC	Manutenzione
TEST_MAC	COLL	
COLL	PROD	Produzione
PROD		
Attività Annullate		

#### Sviluppo in ambiente Consip

Il flusso delle attività comporta la generazione dei nuovi oggetti oppure il prelievo di oggetti esistenti per la loro modifica direttamente sotto il controllo del sistema di

configuration. Le attività di test del software realizzato sono svolte nello stesso ambiente di sviluppo.

#### Sviluppo in ambiente fornitore

L'export delle sorgenti eventualmente necessarie avviene in ambiente di sviluppo.

La consegna delle sorgenti avviene tramite l'inserimento degli stessi nell'ambiente di sviluppo e quindi il loro trasferimento immediato in collaudo. Nel caso di modifica per manutenzione correttiva dovranno, prima del trasferimento in collaudo, essere svolte le attività necessarie per integrare le manutenzioni correttive intercorse.

In entrambe le situazioni le attività e i prodotti (script, ddl, ecc) richieste al fornitore per il trasferimento degli oggetti tra gli ambienti (sviluppo, collaudo, manutenzione, esercizio) saranno specificate in funzione delle caratteristiche dei singoli ambienti predisposti per i progetti.