Business Entity Cloning Engine – SAP NetWeaver Mobile 7.1 Data Orchestration Engine

Applies to:

SAP NetWeaver Mobile 7.1

Summary

This paper describes how Business Entity stored in Enterprise system can be cloned and how individual clones can be managed through SAP NetWeaver Mobile 7.1 solution. It briefly describes the challenges and required capabilities for cloning Business Entities which is critical in mobilizing business process.

Author(s): Ramprasadh Kothandaraman

Company: SAP

Created on: 6th June 2007

Author Bio

Joined SAP in 1999 from then on worked in areas like CRM Middleware, XI and currently working as a program architect in NetWeaver Mobile

Table of Contents

Introduction

Every user in an enterprise needs to have the relevant information at their finger tips to do their daily tasks efficiently. In many cases user may not have the direct connectivity to enterprise system to access the information what he/she needs. Hence there is a need to provision the information on to the edges like Mobile devices which are carried by the user. This information needs to be available whether the user is connected or not connected to the enterprise. Hence there is a strong need to copy or clone the information to the edges. This information in an enterprise system is captured in the form Business Objects or Business Entities. These Business Entities needs to be cloned and should be given to the individual receivers based on their needs. As always this comes with its own challenges. Lets briefly look at some of the challenges posed by Business Entity Cloning and SAP NetWeaver Mobile 7.1 Data Orchestration Engines (DOE) solves this problem.

Challenges in Business Entity Cloning

In this section we will see some of the most important challenges that are posed in Business Entity Cloning. This is not a complete list, but the list of very important challenges that are faced by an engine which does Business Entity Cloning.

- *Creating a clone* – Clone is a kind of entity take away that is going to be given to the end user or a consumer. Obviously every user or system will not need all the information that is stored in an Enterprise System. It will require only the subset of the information which is directed by the needs of the consumer. Needs could be derived out of various sources like Org unit hierarchy, personalization of the user etc., Based on these needs a clone has to be prepared for the consumer. These needs are not static as well, they are quite dynamic. There could be changes in Organization structures, personalization etc., these changes can occur quite often. Also Business Entity is not a static one e.g. Every day thousands of Sales order could be created/updated in the system, these entities needs to be added/updated and distributed to the appropriate clone. This means creation of clone is not the only activity of the engine, it also needs to update the clone depending on changes to the needs. If the engine gives all the information every time then it is not a scalable model, hence there is a need for this engine to provision the real delta.

- *Managing Lifecycle of a clone* – It is clear that clone is an entity take away. Once the set of entities are taken away by the user, these entities will have its own local life cycle e.g. Sales order could be updated on a local device or new Sales order could be created by the end user. All these changes are happening in an environment where there is no direct or synchronous communication to the enterprise system. Final say on the state of an entity remains with the enterprise system. This means that, this engine should be able to bridge the lifecycle of local clone and global enterprise information system.

- *Managing multiple versions of clone* – It is quite possible that one of the consumers has taken a clone on his device and the definition of the entity itself is little bit old e.g. A new field could have been added to the Sales order definition. In this case engine should be able to handle multiple versions of the clone in parallel; essentially it should be able to deal with entity definition gracefully.

- *Recovering lost clone* – It is quite possible that some user might lose the clone e.g. hard disk crash. In this case engine should be able to recover the clone without creating additional load on to the enterprise system which is being used by lots of other users.

- *Channel Agnostic provisioning of clone* – Since clone is close to the end users, it is very important that core capabilities of engine should be independent of the consumption channels and should be open enough to connect to any channels e.g. Some users might prefer RSS feeds for getting notification about their clone, Some other users might prefer a thick mobile client or there could also be another system which is interested in clone.

SAP NetWeaver Mobile 7.1 – Data Orchestration Engine (DOE)

Data Orchestration Engine which is an important component in NetWeaver Mobile 7.1 is the engine which has the capability to create and manage clones for thousands of consumers. Let us briefly look at some of the salient features of DOE which helps to solve the challenges that are described in the previous chapter

- ***Data Consolidation*** – First and foremost, DOE consolidated data from various enterprise systems and stores it in its own local store. Clones are created out of the data that is consolidated. Data is consolidated in the form of entities, not in the form of tables. e.g. Sales order as entity can be defined in DOE and can be connected to various enterprise systems through adapters. Later through this adapter data gets consolidated for various entities. Enterprise system can push the data directly to DOE. DOE also captures the references between entities e.g. Sales order could be referring to customer, this relationship is captured through a modelling tool and at runtime these references are kept intact. This becomes the foundation for creating and managing the clone, also this helps to reduce additional load on the enterprise system. It also helps to decouple clone management logic out of the enterprise system.

- ***Data Distribution*** – Second and one of the most critical aspect is the creation and management of clones. This is done by data distribution module. This module takes rules and dependencies which are defined at design time and automatically manages the subscription for the clone consumers e.g. For distributing Sales order based on region a simple rule can be defined like Sales Order.Region = Consumer.Region. Then automatically whenever consumer is created a subscription will be generated for that consumer based on region attribute and also whenever region attribute changes, subscription will also change. Once the subscriptions are managed, then this module calculates who needs to get what and maintains this information on a store. This store keeps track of who needs to get what and also who has got what. Rule is not the only way to manage the clone, it is also possible to define dependencies e.g. We can define a dependency between Sales Order and Customer in such a way that, a referred customer should also be part of the clone if the Sales Order becomes relevant for the clone. Since this module maintains the inventory of who gets what information, recovering this information becomes a straight forward activity. The engine can now easily recover a lost clone.

- ***Receiver inventory management*** – Clone is created for some receivers. DOE has a receiver inventory where it tracks all the consumers of the clone and it also maintains various attributes of the receivers based on which subscriptions can be automatically managed. This inventory is a flexible one in a way that, depending on the scenario inventory definition can be enhanced to hold additional attributes. Also this inventory can be automatically managed by providing an entity and an adapter to the appropriate source e.g. For every employee in an organization, one receiver can be automatically generated along with all their attributes by consuming the employee information from the HR system.

- ***Bi-directional entity updates*** – It is clear that clones are owned by individual consumers and clones can have its own lifecycle. In a clone entities can be created, updated or deleted. But this has to be validated with an enterprise system. In this case DOE acts as a bridge, where every clone synchronizes its changes with DOE and DOE will respond back with a confirmation or rejection for the changes. If the changes are rejected then it can be resolved either by clone owner or by a central administrator.

- ***Multiple Version Handling*** – DOE allows multiple versions of Entity definition to be active in parallel. It can get/put information from/to a clone in its own local version and later will bridge it with the actual version of enterprise system.

- ***Pluggable Channel Handlers –*** End user might be used to view the information through various means e.g Some users might want to see it through RSS readers and some others might be interested in doing their activities on a thick client like PDA or Laptop. In order to support these kinds of multiple consumption channel, outbound and inbound side of DOE is opened up to plug-in channel specific handlers which can render the information in an appropriate way e.g. For RSS, one RSS outbound channel can be plugged in which can create RSS feed XML out of the clone identified for the receiver.

Copyright