ARCHITECTURE PATTERNS FOR ENTERPRISE IDENTITY MANAGEMENT WITH MICROSOFT IDENTITY INTEGRATION SERVER

A WHITE PAPER

6th September 2004
Version: V2.1

Gerrit van der Geest
Evan Erwee

Navit (Pty) Ltd
Navigating in information technology

CONTENTS

1. INTRODUCTION 3

1.1. Identity Management 3
1.2. The Enterprise environment 4
1.3. Microsoft Identity Integration Server 5
1.4. Microsoft Active Directory 6
1.5. Microsoft Active Directory Application Mode 7

2. PATTERNS FOR IdM IN THE ENTERPRISE 7

2.1. The example 7
2.2. The architecture pattern on forest level 8
2.3. The architecture patterns on Enterprise (inter-forest) level 9
2.4. Meeting the requirements 13

3. EXAMPLES OF APPLICATION OF THE PATTERNS 15

3.1. Enterprise GAL 15
3.2. Self help 16
3.3. Object life cycles 17

4. CONCLUSION 18

1. INTRODUCTION

This white paper will introduce architecture patterns for the implementation of Identity Management in the Enterprise. The patterns will be based upon the functionality as provided by the Microsoft Windows Server 2003 integrated product set.

1.1. Identity Management

Identity Management (IdM) according to the Burton Group can be defined by:

"A set of processes, and a supporting infrastructure, for the creation, maintenance, and use of digital identities."

In this white paper we will focus on the supporting infrastructure but it is our experience that the introduction of IdM without the implementation of the governance and administrative procedures is like implementing Microsoft Active Directory without tuning the service organisation. Enthusiastic start followed by graceful degradation.

DIGITAL IDENTITY VERSUS PERSONAL IDENTITY

In the majority of the literature and by many of the vendors, "digital identities" are immediately translated to "personal or individual identities". Within this step, a significant aspect of IdM is completely ignored. Personal identities can not (yet) interface with the IT world without making use of some form of physical device which, in itself, is also associated with a "digital identity". The developments in the last decennia demonstrate that the physical devices and the way they interact with the IT infrastructure play a dominant role within all initiatives to increase the security level and that their management account for a considerable cost component. Reduction of security risks and costs are main business drivers behind an IdM initiative and therefore these digitalised identities must not be neglected at all.

The digital identity of a person (human identity) is reflected by instances of multiple classes (multiple objects) in IdM. This is illustrated in the picture below. User (account) objects, workstation objects, group or role objects, server, service, application objects, device objects etc. all play their role in an IdM reflection of a human identity.

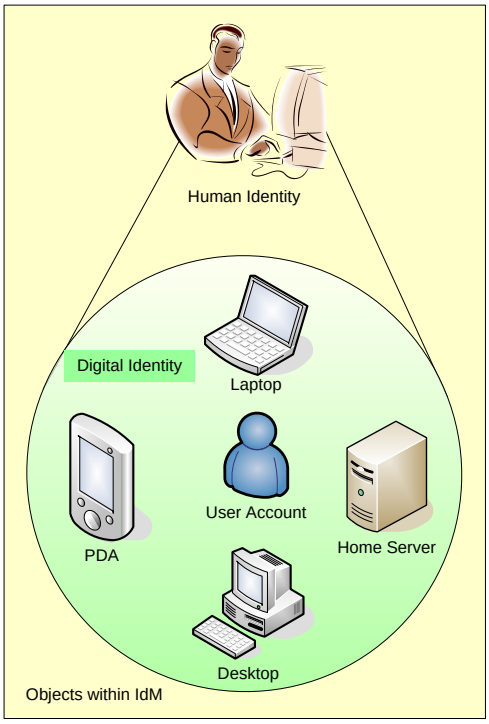

Figure 1 Digital identity

Enterprise IdM versus Federated IdM

In this white paper we will focus on Enterprise IdM (intra-enterprise) and not deal with Federated IdM (inter-enterprise). At this stage, these two forms of IdM are dealt with as nearly being separate worlds. A different set of standards and products are applied for each. However, it is expected that these two worlds will synergise in the near future. Is it possible anyway, to deal with federated identity management if Enterprise identity management is not yet catered for?

Architecture pattern

In this white paper we will not go into detail on all the possible application areas of IdM such as:

- object provisioning and decommissioning;
- object life cycle models;
- delegation of administration and its most extreme form of self service;
- providing a consolidated and valid view of a digital identity;
- risk and compliancy management.

Instead, we will discuss architecture patterns for the implementation of IdM in an Enterprise that can ultimately provide services for all application areas mentioned above.

The next section will deal with the challenges that have to be met when applying IdM in an Enterprise environment.

1.2. The Enterprise environment

An Enterprise environment brings in specific challenges regarding an identity management solution. Some of these challenges are:

- the Enterprise may consist of highly autonomous operating business entities;
- the Enterprise may be geographically spread across the world and will be faced with legal restrictions regarding cross border information exchange;
- the Enterprise needs a consolidated view of a subset of all the objects that are hosted within several businesses;
- the authoritative source for the identity information will in some cases be centralised but may also be distributed and even a combination of these two scenario's will exist. For example a centralised HR system and decentralised switch-board systems;
- a subset of the Enterprise identity information needs to be available within the several businesses to facilitate functionalities like an Enterprise wide Global Address List, and authentication and authorisation for an Enterprise wide intranet;
- the Enterprise will constantly be in a flux regarding its IT infrastructure, partly due to mergers and de-mergers. It will be faced with a non-consolidated, non-rationalised and heterogeneous infrastructure.

These challenges can be translated to the following requirements regarding the architecture to be implemented. The IdM solution must:

- be able to exchange information with products of different brands, different versions and must support at the minimum the relevant industry standards like LDIF and DSML;
- not be tightly bound to a specific Network Operating System;
- support a highly distributed environment and must offer mechanisms for the required information distribution;
- support a multi-master replication concept in which changes are initiated from distributed locations;
- be able to provide a consolidated view when needed but it may not be based on a concept in which all identity data is completely distributed;
- provide mechanisms to optimise the uncoupling. The functionality of the collective may not be completely dependent on the well-being of all constituent elements. And the functionality of the constituent elements may not be completely dependent on the well-being of the collective;
- be able to operate while not violating existing service/data isolation and autonomy boundaries;
- not compromise the existing security levels.

We will focus in this white paper on the integrated product set of Microsoft that consists of Windows Server 2003 that includes Active Directory (AD) and Active Directory Application Mode (ADAM), and the

additional product Microsoft Identity Integration Server 2003 (MIIS). In the next section we will position these products briefly in the context of Enterprise IdM. Subsequently architecture patterns for Enterprise IdM based on this product set will be discussed that will fulfil the above stated requirements.

1.3. Microsoft Identity Integration Server

Microsoft Identity Integration Server is a lifecycle management system that supports provisioning of identities, password management and keeps connected data sources (directories and other information stores like databases and flat files) in sync through its metadirectory functionality. It is important to understand that MIIS is not a directory service in itself. It is a synchronisation service. The information stored in MIIS only serves the objective to synchronise the information with connected data sources. As a consequence, there is no client access to update information in MIIS and MIIS does not provide an LDAP compliant interface.

MIIS consists of the following major components:

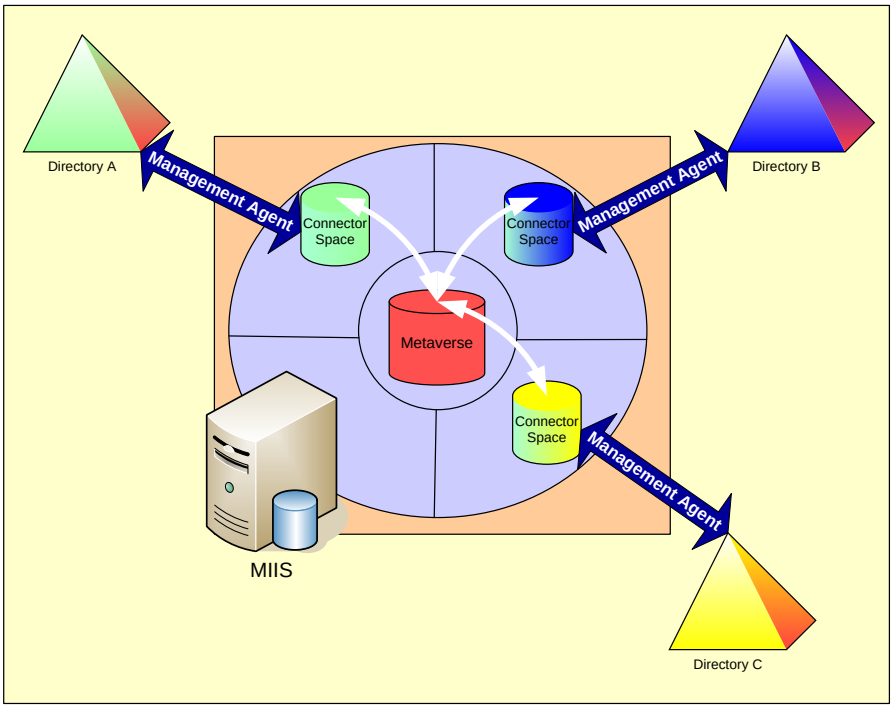

Figure 2 MIIS Components

- a database (metaverse) that will contain the consolidated state of all relevant objects. The metaverse is implemented as a table in Microsoft SQL Server. In the above picture the metaverse will contain the consolidated view of Directory A, B and C.
- management agents that link a connected data source to the metaverse. The management agents direct the information flow between the connected data sources and the metaverse. The management agent is configured by means of rules to manipulate the data flow from a connected data source, via the connector space to the metaverse and vice versa.
- connector space which contains logical storage areas used by the management agents to stage the information flow between the connected data source and the metaverse. The connector space is also stored within Microsoft SQL Server. Depicted above as a green, blue and yellow database.

MIIS provides management agents for several types of connected data sources like Network Operating System (NOS) directories, Messaging platform directories, LDAP compliant stores, (application) databases and flat files like LDIF, CSV and DSML files.

A much more elaborate description of MIIS can be found in [1] and [2].

For our discussion it is important to realise that:

- changes from MIIS to connected data sources can only be triggered by a change in the metaverse;
- changes to the metaverse can only be triggered by a change in the connector space of one of the management agents;
- changes to a connector space can only be triggered by changes in the associated connected data source;
- MIIS is a pure state based engine. The state of the connected data source is mirrored in the connector space. The connector space provides mechanisms to keep track of changes in the connected data source and of outstanding changes still to be applied to the connected data source. This mechanism, which is slightly hidden from the outside world, makes MIIS a very strong candidate for our purpose because it provides the required uncoupling.
- MIIS provides a WMI compliant interface mainly aimed for the management of MIIS;
- MIIS will provide in the future an additional graphical viewer (Polyarchy Preview) into the aggregated identity information as it is stored in the metaverse. Polyarchy Preview is not yet released.
- The MIIS functionality is currently packaged in a full functionality package (Microsoft Identity Integration Server 2003) and a version called the Identity Integration Feature Pack (IIFP) which offers the same functionality but is restricted to AD and ADAM management agents only. The IIFP is for free.
- MIIS inherits scalability from the SQL Server based implementation. MIIS does not provide scalability over a distributed architecture but, as we will show in the architecture patterns, that is not needed.

1.4. Microsoft Active Directory

The positioning of Active Directory in the infrastructure architecture is changing. In the past it was positioned as the central directory hosted within the NOS and providing services to all infrastructure components. Active Directory provides the possibilities to extend the schema and, as from the Windows 2003 release, it also supports application partitions. Thereby it can fulfil the central role. However the current trend is towards an architecture based upon multiple directories. The rationale for this is found in security isolation, autonomy both from administrative point of view but also from product point of view, segmentation in order to reduce risk impact and maybe also by the fact that the implicit loss of integration can be compensated for by means of solutions like IdM.

The trend can be recognised in the following facts:

- the positioning of single versus multiple forests has been changed considerable in Windows 2003;
- Active Directory 2003 provides more enhanced facilities for inter-forest trust by means of the forest trust and by means of the external trust and the increased security configuration options by the introduction of the selective authentication model;
- the release of ADAM as will be discussed in the next paragraph;
- the positioning of MIIS as the facility for cross forest synchronisation.

Observing this trend, you may even argue if the tight integration of Microsoft Exchange with Active Directory is the best way forward.

In our Enterprise scenario we will position Active Directory as the NOS directory. It is assumed that within the Enterprise multiple Active Directory forests will be implemented. In the architecture patterns we will focus on the internal network but as described in the Microsoft Systems Architecture, additional Active Directory implementations will need to exist within the perimeter networks to provide the required infrastructure security and manageability.

We follow in our approach the principle:

"Active Directory is the NOS directory service and the strategy is to keep Active Directory as lean as possible. The Active Directory schema will only be extended for those fields needed in order to play its role as information provider/consumer for the IdM solution."

1.5. Microsoft Active Directory Application Mode

Microsoft Active Directory Application Mode (ADAM) is part of Microsoft's integrated directory service available with Windows Server 2003. ADAM provides a full LDAP compliant directory service that does run as a non-operating system service and thereby can be regarded as an application of which multiple instances can be implemented within one operating environment.

ADAM inherits the main properties from Active Directory. A full multi-master replication model as implemented within Active Directory is available in ADAM. This replication model will play a significant role in one of our architecture patterns.

An ADAM instance may contain one or more configuration sets and each configuration set will consist of multiple partitions:

- a schema partition that holds the definition of classes and attributes;
- a configuration partition that contains the configuration data concerning partitions and replication;
- one or multiple application partitions that will contain the application data.

The replication mechanism will copy ADAM directory data updates that are made to a directory partition on one ADAM instance, to other ADAM instances that hold copies of the same directory partition. ADAM instances that hold copies of the same directory partition or partitions form a logical grouping called a configuration set.

There are some more interesting features offered by ADAM that may play a role in an Enterprise IdM solution. These are:

- ADAM supports the LDAP referral mechanism as defined in RFC 2251 by which ADAM can point clients to another location for LDAP queries. This can be useful for example to refer LDAP clients to an Enterprise wide GAL.
- applications that cannot authenticate against Active Directory can make use of bind redirection as offered by ADAM to authenticate against ADAM.

2. PATTERNS FOR IdM IN THE ENTERPRISE

Having all these features of these products, how can they be combined in such a way that they can fulfil the requirements of an Enterprise IdM solution?

In this section architecture patterns are defined to address these requirements. The first pattern (no. 1) will address the requirements within the context of a forest (for example business unit or country). To cater for the inter-forest Enterprise requirements, two additional Enterprise patterns (no. 2, 3) will be discussed. Dependent on the situation at hand, a specific pattern or combination of patterns can be applied and be tailored to the specific needs.

2.1. The example

As an example, let us use an Enterprise that has a corporate head office in South Africa (ZA) and has subsidiaries all over the world and let's focus on some of them, one in the United Kingdom (UK) and one in New-Zealand (NZ). The Enterprise maintains information about the employees in HR systems localised in head office and one per country. This solution has been chosen to cater for country specific HR requirements. They need an Enterprise wide global address list and corporate wants to have a consolidated view of all employees for security reasons.

The Enterprise implemented successfully Windows Server 2003 as their NOS. The security requirements, strong autonomy of the business entities and legal requirements made them decide to implement a multi-forest Active Directory environment. They have implemented one forest for corporate and one forest per country. There was no need for any resource sharing between the forests and therefore no external or forest trusts exist. The Enterprise has chosen Exchange 2003 as their messaging platform and per forest one Exchange organisation has been implemented.

Note that the fact that the Enterprise has chosen to implement a forest per country is by no means a general recommendation. The most favourable situation is still a one forest situation followed by segmentation based on service and data isolation requirements.

Corporate security policies force the implementation of user object life cycle models to ascertain that only user accounts exist for those employees that do have an active employment. As soon as an employee leaves the organisation, the user account must be blocked for further access or even be deleted. In addition, they want to implement a workstation object life cycle to ascertain that only workstations that are still "in use" will be allowed to authenticate and that personal workstations that belong to employees that have left the organisation can no longer authenticate.

2.2. The architecture pattern on forest level

Per location (ZA, UK and NZ) they have implemented an AD forest. There are no trusts established between the forests for the following reasons:

- they do not need resource sharing between the forests;
- they want to keep the isolation and autonomy boundaries at the forest level.

PATTERN 1: IDENTITY MANAGEMENT ON COUNTRY (FOREST) LEVEL

The figure below shows a simplified representation of the implementation per country.

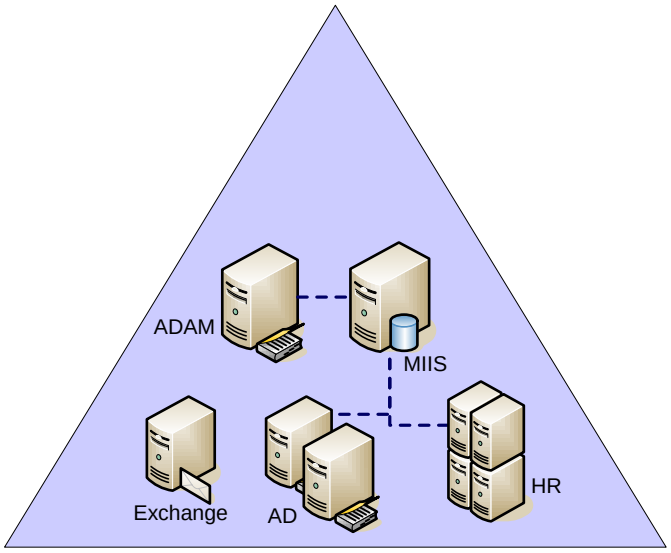

Figure 3 Implementation per country

Within the forests, member servers are implemented for the following purposes:

- Human Resource application;
- Exchange (one Exchange organisation per forest);
- ADAM directories;
- MIIS application, including the required SQL Server databases.

This is a highly simplified representation because there will be file, print, database servers and so on, but they do not play an important role in our discussion.

The ADAM directory will serve in our example in essence the following purposes:

- it provides a directory partition for all the identity information that is not native in one of the other systems;
- it provides an application interface via a writeable partition to the IdM solution. Typical examples of applications that will make use of this mechanism are:

- a web application for the maintenance of white page information;
- self service applications;
- password synchronisation services.

- it provides a platform to deliver authentication, authorisation, administration and auditing services for applications that cannot be integrated within Active Directory.

The MIIS implementation within the forest will cater for the required synchronisation of the identity information as contained with Active Directory, the several ADAM directories and systems like HR. The picture below illustrates the logical structure of the MIIS implementation within the forest.

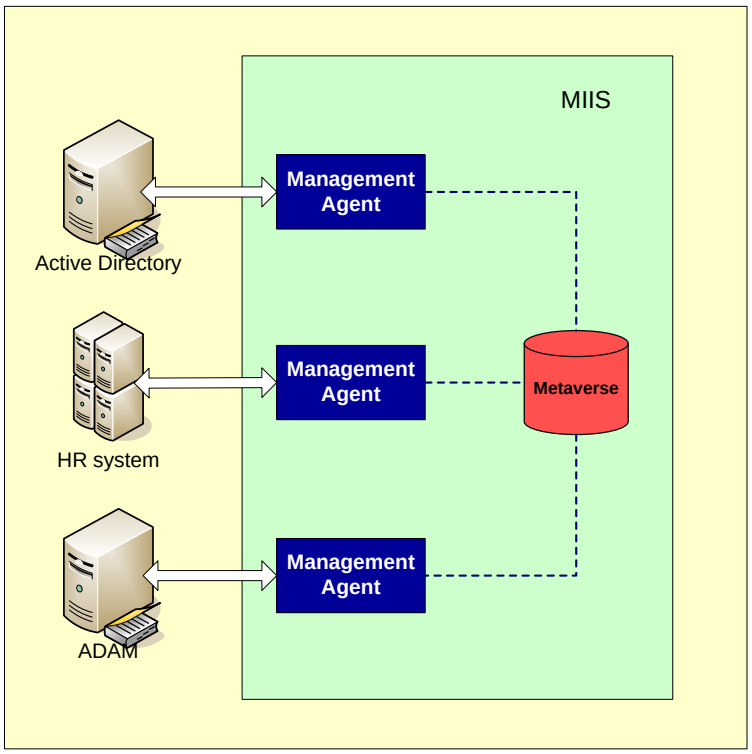

Figure 4 MIIS Logical structure

This structure will provide for all functionality to cater for:

- implementation of object life-cycle models;
- administrative delegation;
- object provisioning and decommissioning;

on country level.

2.3. The architecture patterns on Enterprise (inter-forest) level

The pattern described in the previous paragraph (no. 1) caters for the required IdM functionality on country level but does not provide the required Enterprise wide (inter-forest) functionality. In this paragraph we will discuss two additional patterns so-called Enterprise patterns (no. 2 and 3) to cater for the Enterprise (inter-forest) requirements.

The most obvious solution seems to be to implement an Enterprise MIIS located in for example South Africa that will have connected data sources throughout the Enterprise. It will have management agents for all Active Directory forests, all HR systems and all ADAM partitions.

This solution has many disadvantages, but the most obvious one is that it will result in intensive data transport. The majority of the identity information only has a localised, country specific meaning while in this scenario all of that will go Enterprise wide. This is not the recommended way forward.

We have to come up with a better solution to meet the requirements. In fact there are two quite similar patterns that can be applied. They share the following concepts:

- the implementation per country is still based on the pattern depictured in Figure 3. Additional components are added to cater for the Enterprise wide functionality;
- the required consolidated identity information on Enterprise level is stored in a separate ADAM Enterprise application partition;
- dedicated country specific subsets of the identity information are created that will contain the identity information to be synchronised and shared on Enterprise level. They are also stored in ADAM application partitions. The use of the country specific partitions with subset of data offers two main advantages:
  - strong reduction of the amount of information to be replicated;
  - limitation of the risk that conflicts arise due to country specific regulations and to regulations regarding information storage and distribution.
- the additional application partitions are hosted in additional ADAM instances (at least additional configurations sets) and do not share the ADAM instance which is part of the country (forest) level (see Figure 3) in order to be able:
  - to delegate administration to different roles within the Enterprise;
  - to implement data isolation between country and Enterprise level;
  - to have different schema and application partitions.

  In the pictures below that will illustrate the two patterns, the application partitions are illustrated as cylinders. The colour of the cylinder reflects if it is the Enterprise application partition (yellow) or one of the application partitions that host a country specific subset (colour of the country).
- the Enterprise application partition and the several country specific application partitions are synchronised by means of an MIIS implementation as illustrated in Figure 5.

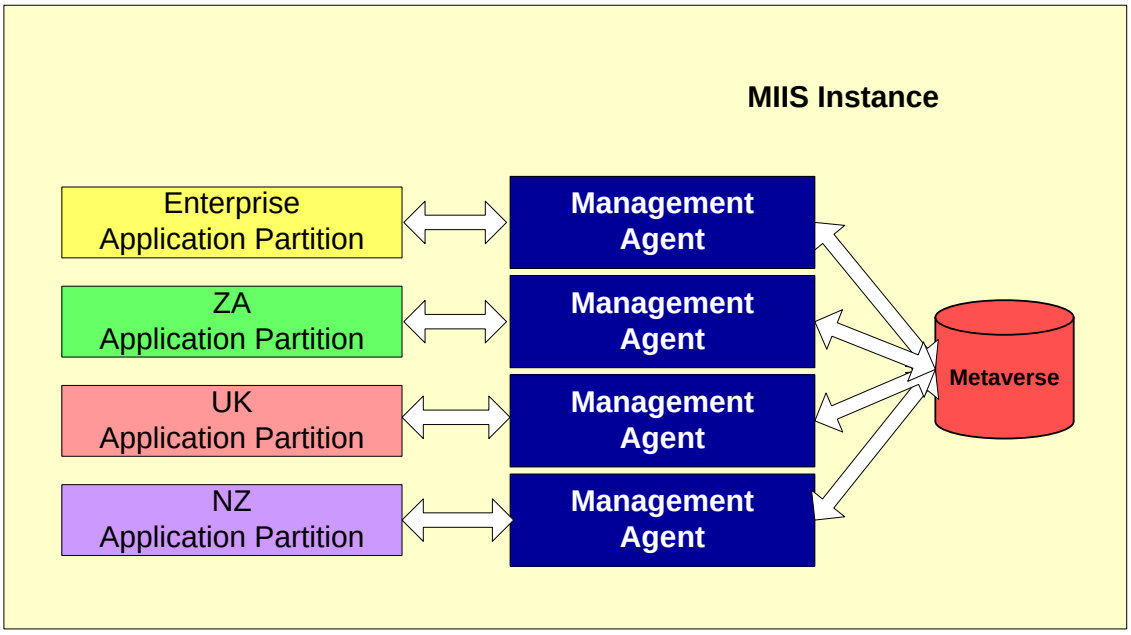

Figure 5 Synchronisation of partitions by MIIS

Pattern 2: Decentralised country specific application partitions

The architecture pattern that is based on decentralised country specific application partitions is illustrated in Figure 6.

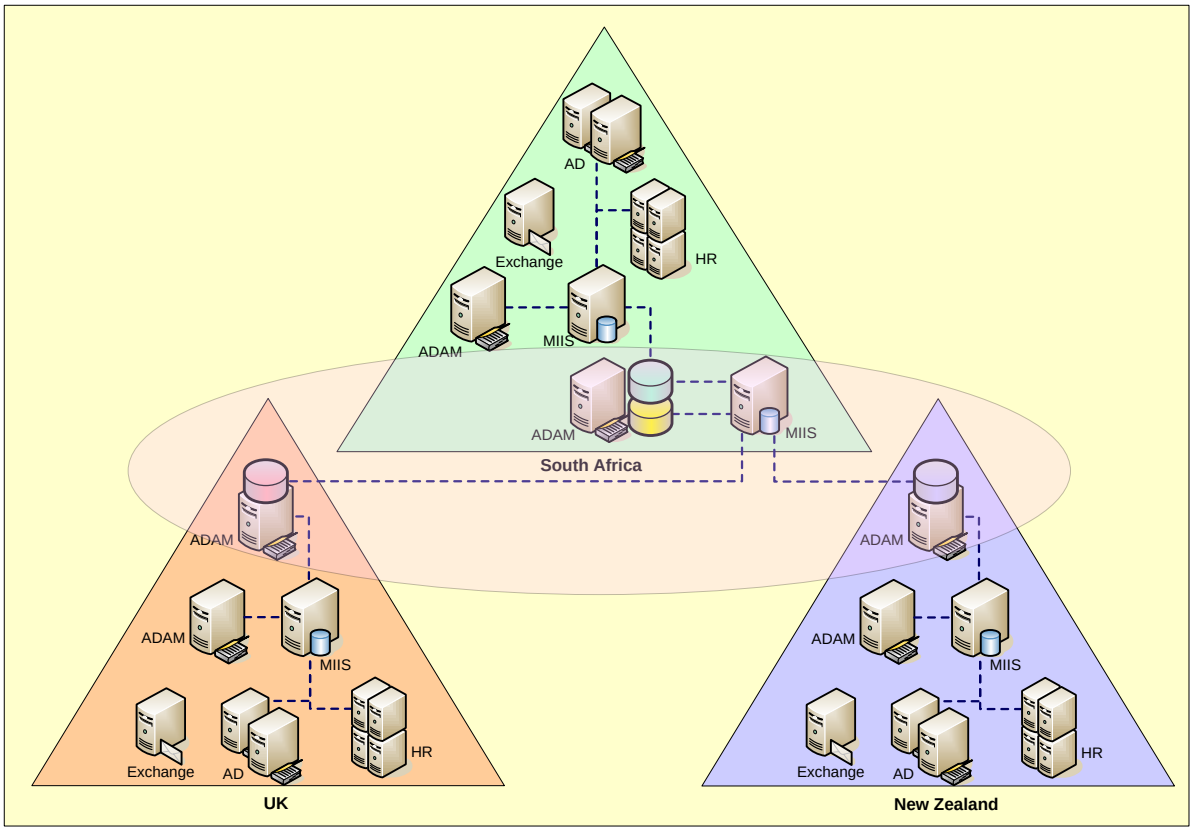

Figure 6 Decentralised country specific application partitions

Per country an ADAM instance has been added that will contain the country specific application partition. The ADAM instance in South Africa (Head Office) will also contain the Enterprise identity partition.

The ADAM configuration needed for this pattern is depictured below:

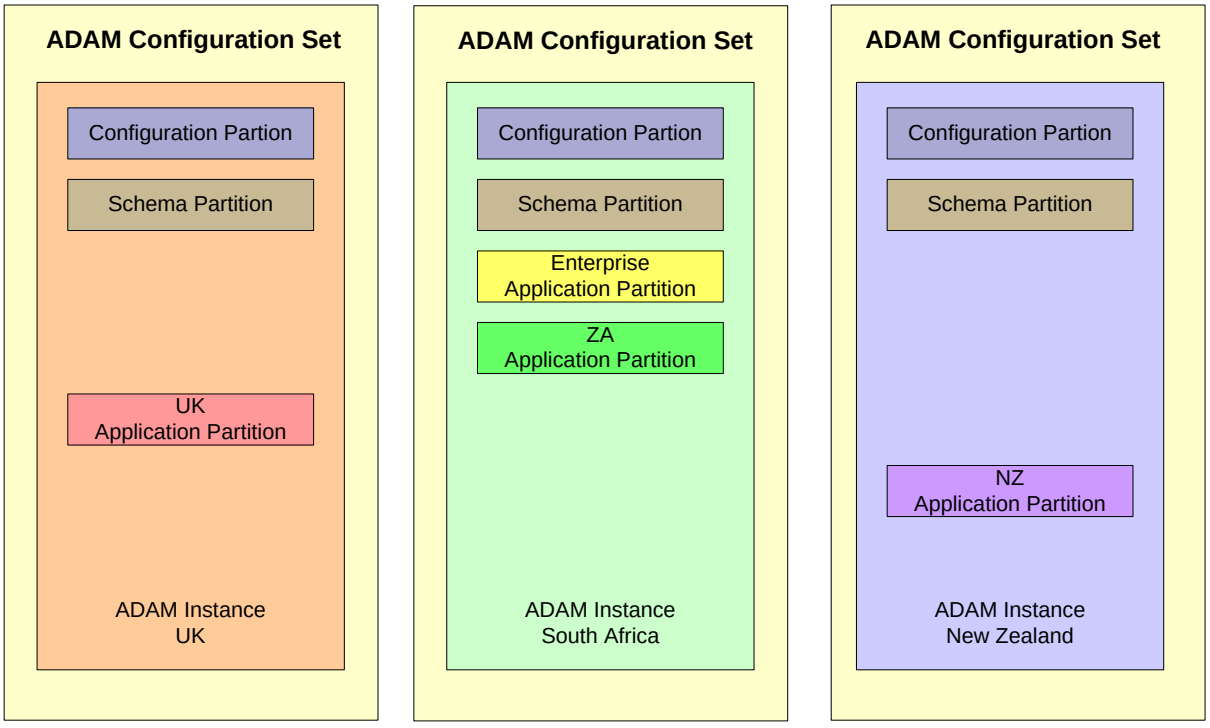

Figure 7 ADAM configuration for pattern 2

In South Africa an additional MIIS implementation caters for the Enterprise wide identity consolidation and aggregation. It will have one management agent for the Enterprise application partition and one management agent per country application partition. For this MIIS implementation, use can be made of the Identity Integration Feature Pack.

The MIIS implementations that form part of the country implementation will be extended with one management agent to synchronise the country specific Enterprise application partition with the other connected data sources.

If due to some reason it is not possible to host a country specific partition in one of the countries, that partition may be hosted in the ADAM instance in Head Office (South Africa).

Pattern 3: Centralised and decentralised country specific application partitions

The architecture pattern that is based on centralised and decentralised country specific application partitions is illustrated in Figure 8. In this pattern the advanced functionality of ADAM for partition replication is used to the full extend.

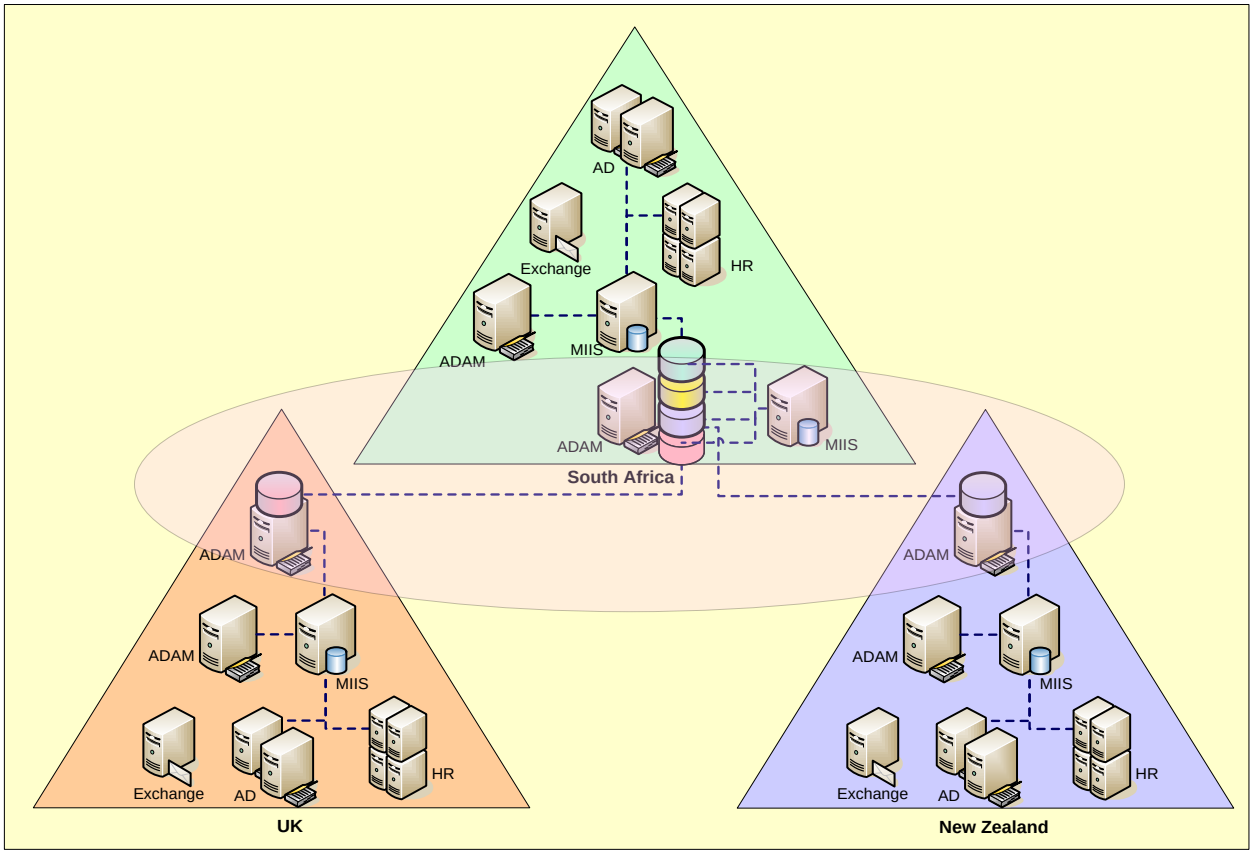

Figure 8 Centralised and decentralised country specific application partitions

The added ADAM instances within the countries are made part of an Enterprise wide ADAM configuration set. Within a configuration set, partitions can be synchronised by means of the multi-master replication model of ADAM. The country ADAM instance will contain a partition (or maybe even multiple partitions as explained later) that contains the identity information that has Enterprise wide meaning.

The ADAM configuration set will have a structure as illustrated below:

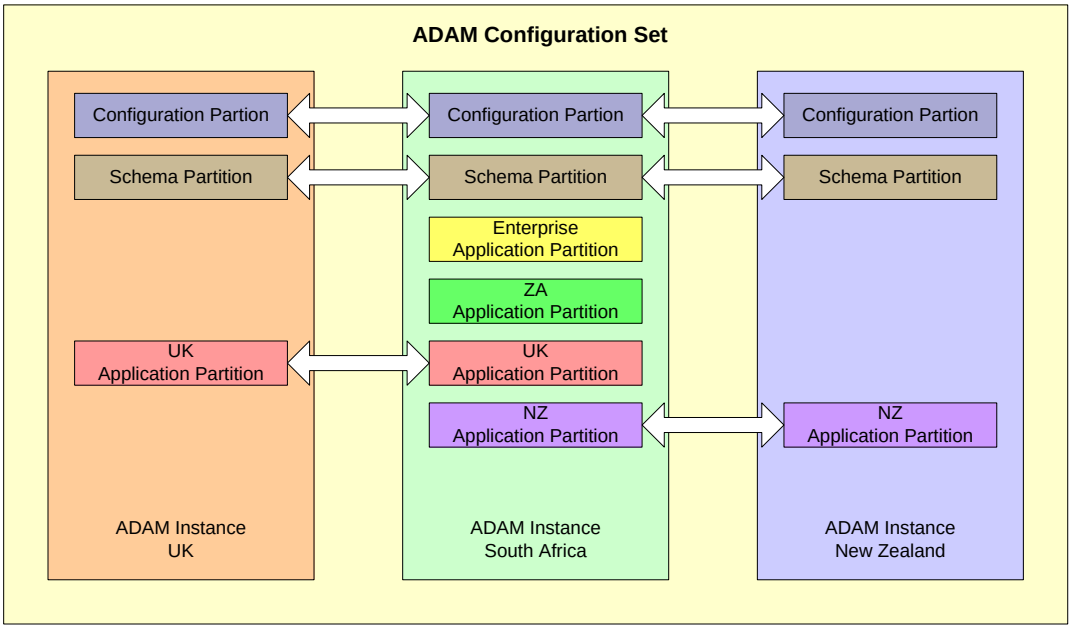

Figure 9 ADAM configuration for pattern 3

The schema and configuration partition are shared between the countries ADAM instances. The country specific application partition will be replicated to the ADAM instance on Head Office (South Africa) level. The ADAM instance on Head Office level will contain replicas of all the country specific partitions plus the so called Enterprise application partition. The Enterprise application partition contains the consolidated view of the identity information on Enterprise level.

2.4. Meeting the requirements

In this paragraph we'll provide an overview on how the requirements as stated in paragraph 1.2 are met by applying pattern 1 together with any of the Enterprise patterns (no. 2, or 3) or a hybrid of these.

The situation at hand will dictate the most appropriate pattern. The main difference between pattern 2 and pattern 3 lies in the fact that in pattern 3 the ADAM multi-master replication model caters for the information distribution. This provides extra facilities like data compression and a very rich configuration model with embedded redundancy options.

The IdM solution must be able to exchange information with products of different brands, different versions and must support at the minimum the relevant industry standards like LDIF and DSML.

This requirement is met by the standard functionality available in MIIS. MIIS provides management agents for a broad spectrum of connected data sources (see for a recent list the Microsoft web site on MIIS) and it is expected that the number of supported management agents will increase further in the near future. MIIS supports the LDIF and DSML v2.0 standards.

The IdM solution must not be tightly bound to a specific Network Operating System.

The MIIS and ADAM instances need to be installed on a Windows platform. They do not need to be installed on a Windows domain member server however this will have a negative impact on the security level and the functionality. In addition, the Enterprise wide synchronisation is made independent of an Enterprise wide Active Directory forest.

In the situation that one of the countries based their NOS on a different platform, lets take Novell as an example, the same pattern may still be used. The alternatives are:

- implement an MIIS (not member) and an ADAM server in that country and configure MIIS to synchronise with the Novell NOS directory;
- dedicate the local identity management function to, for example a Novell Nsure Identity Management implementation, and add an ADAM server.

An example by applying pattern 1 and 3 is given in the picture below.

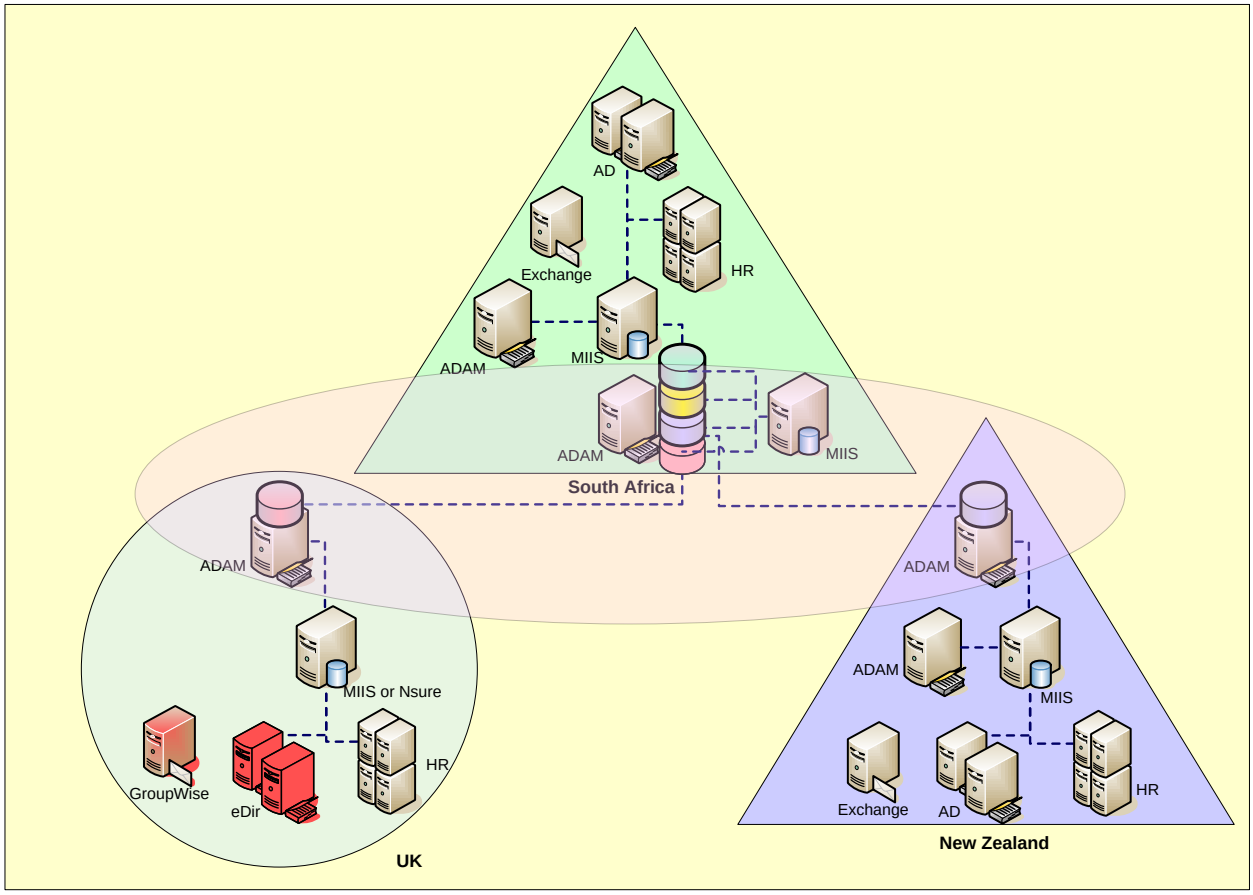

Figure 10 Integration with other platforms

The IdM solution must support a highly distributed environment and must offer mechanisms for the required information distribution.

MIIS or the ADAM multi-master replication model caters for the required information distribution. In addition, the segmentation of the identity information in several partitions provides for optimisation of replication traffic and can be optimised to prevent legal conflicts regarding cross-border information exchange.

The IdM solution must support a multi-master replication concept in which changes are initiated from distributed locations.

In pattern 2 the multi-master replication is catered for by means of the MIIS implementation. In pattern 3 the ADAM multi-master replication model combined with the MIIS functionality provides the required functionality.

The IdM solution must be able to provide a consolidated view when needed but it may not be based on a concept in which all identity data is completely distributed.

The consolidated view is offered by means of the Enterprise application partition in ADAM and the Enterprise metaverse in MIIS. It does contain the consolidated view of only those objects and attributes

that are considered to be relevant on Enterprise level. The majority of the objects and attributes will not be replicated to Enterprise level but will stay on country level.

The IdM solution must provide mechanisms to optimise the uncoupling. The functionality of the collective may not be completely dependent on the well-being of all constituent elements. And the functionality of the constituent elements may not be completely dependent on the well-being of the collective.

The uncoupling is achieved by:

- the use of the additional ADAM application partitions. Within a country the local functionality is isolated from the functionality on Enterprise level by using dedicated application partitions for the Enterprise functionality;
- the embedded functionality offered within the MIIS management agents as described in section 1.3.

The IdM solution must be able to operate while not violating existing service/data isolation and autonomy boundaries.

The management agents that interface with Active Directory will need to have access to the relevant objects. The credentials to be used by the management agents can be defined. Privileges comparable to those delegated to a data administrator will be needed. The scope within Active Directory can be limited for the management agent by defining a partition (domain) and subsequently the specific containers (OU's) to be accessed.
The fact that additional ADAM partitions are being used to provide the Enterprise functionality makes it possible to isolate the administrative responsibilities between Enterprise and country level.

The IdM solution must not compromise the existing security levels.

The LDAP traffic between the MIIS management agent and Active Directory can be digitally encrypted by using the Kerberos authentication protocol providing MIIS is hosted on a domain member server.

The traffic between MIIS and the ADAM partitions can be secured by encryption and signing.

Regarding pattern 3: to ensure replication security, ADAM authenticates replication partners before replication, and replication authentication always occurs over a secure channel. There are three different levels of replication authentication. In the proposed pattern the ADAM instances are not within the same forest and no trust exists between the forests. As a consequence the only replication authentication level that is possible is: "Negotiated pass-through". This requires that the services accounts of the several ADAM instances within the configuration set do have the same name and the same password.

3. EXAMPLES OF APPLICATION OF THE PATTERNS

In this chapter three examples are discussed that will show the application of these patterns.

3.1. Enterprise GAL

One of the most obvious reasons to implement an Enterprise IdM solution is the fact that Enterprises want to be able to use an Enterprise wide Global Address List to support for example messaging infrastructures. The countries will have to make their local GAL information available for Enterprise wide synchronisation and consolidation.

Within the patterns discussed, the Enterprise GAL information may be contained within the Enterprise application partition or may even be located in an additional application partition as reflected in Figure 11.

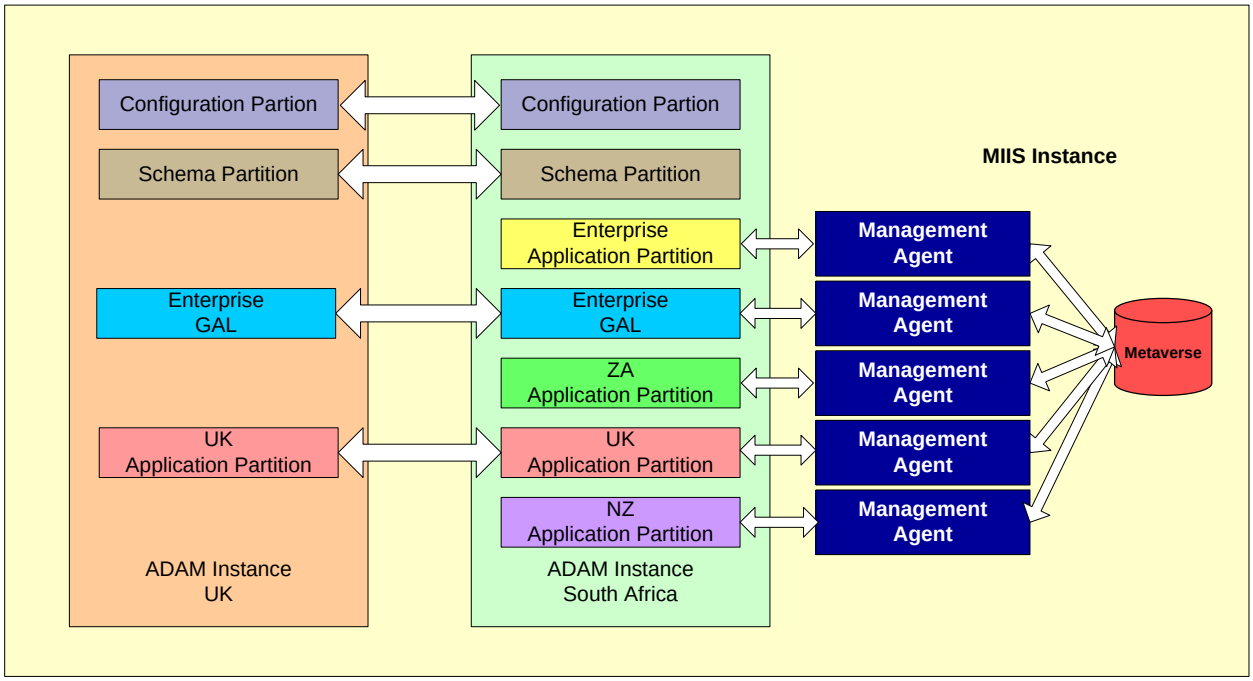

Figure 11 Enterprise Global Address List

3.2. Self help

Self help is the ultimate form of administration delegation. By means of self help a user is enabled to:

- maintain his/her own white page information;
- reset his/her password;
- maybe unlock his/her account;
- perform some form of entitlement management.

The self help functionality is offered by means of a form of a web based application that executes with the user credentials or alternatively some form of a kiosk solution is used to provide initial access with default credentials. Self help functionality is in most cases combined with a facility to synchronise the information across multiple platforms/applications. Changes made in one system are replicated to other systems.

In the architecture patterns presented in this white paper the self help application will interface with a writable partition in ADAM. Subsequently the information will be synchronised by means of MIIS to the subscribed information systems.

The picture below shows a logical representation of a self help scenario:

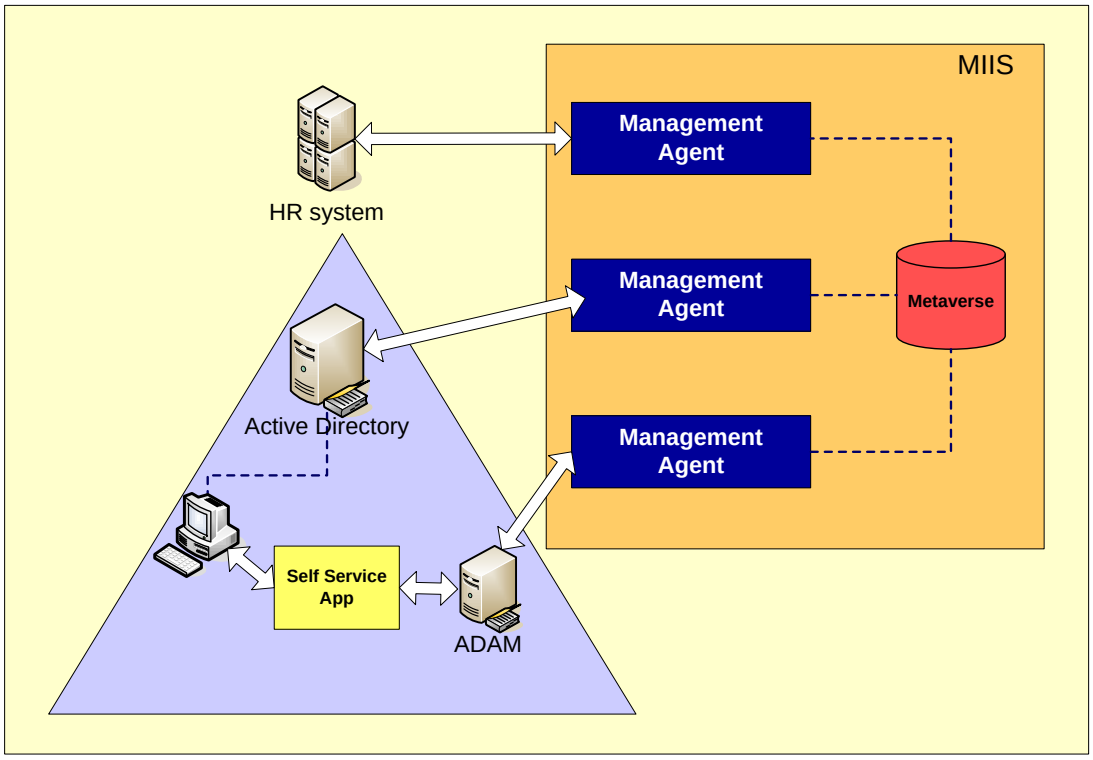

Figure 12 Self help

Let us take an example in which the user needs to update some white page information. The definitive sources of that information are partly located in the HR system and partly in Active Directory. The user will update the white page information in a writeable ADAM partition and MIIS will subsequently cater for the synchronisation of that information to the HR system and to Active Directory.

In order to make this happen, the user will first authenticated against Active Directory. Subsequently, the user will have to access an ASP.NET Web application and Windows integrated authentication is being used by that application. The application will retrieve the existing information from the ADAM partition which is kept synchronised by MIIS with the HR and Active Directory systems. In order for the application to be able to access the data in the ADAM partition, it will perform an LDAP bind based on Kerberos authentication. The user will make the required updates and the updates will be stored in the ADAM partition and subsequently be synchronised to the HR and Active Directory systems.

Subsequently the Enterprise identity system will synchronise the HR and Active Directory (being the definitive sources) via the country ADAM partition with the Enterprise identity store when needed. Any of the three patterns discussed may be applied.

3.3. Object life cycles

The implementation of an integrated object life cycle model throughout the Enterprise may contribute significantly to risk reduction and provides the foundation for facilities like entitlement management and administration delegation. The object state may be dependent on events that are captured throughout the Enterprise and must be available throughout the Enterprise. The architecture patterns presented in this white paper provide the mechanism for the implementation of Enterprise object life cycle models. The object state will be maintained on country level by applying pattern 1 illustrated in Figure 3. Subsequently the object state may be distributed to Enterprise level by using any of the other two patterns. The consolidation of object state on Enterprise level may be required to block for example access to Enterprise wide extranet solutions dependent on the state of the object.

4. CONCLUSION

The architecture patterns as presented within this white paper are capable to meet the stated challenges for Enterprise IdM. Enterprises on their journey to Enterprise IdM should consider the application of such patterns in order to fulfil their requirements on the long term. Many initiatives that currently take place should be considered as creating point solutions and not as the first step towards a strategic way forward. It is expected that the functionality offered by the constituent elements like MIIS will increase and new facilities will be offered for example in Active Directory to pave the way for the implementation of password change self help and entitlement management. Enterprises that have implemented a proven architecture will be capable to benefit in an early stage from these new developments.

The introduction of Enterprise IdM must start with the implementation of the correct architecture and the establishment of an appropriate governance and administrative organisation. Aim for meeting business drivers that provide a fast return on investment.

REFERENCES

[1] Essential Concepts for Microsoft Identity Integration Server 2003, www.microsoft.com.

[2] Microsoft Identity Integration Server 2003, A white paper by Oxford Computer Group. See www.oxfordcomputergroup.com.

ABOUT THE AUTHORS

Gerrit van der Geest

Gerrit van der Geest is managing director of Navit (Pty) Ltd, a South African based company specialised in Enterprise IT Architecture and Project Management. After completing his study Applied Physics in the Netherlands, Gerrit was employed by Philips and Digital Equipment before establishing his own IT company. Within his 20+ years of experience, he has built up an extensive track record in enterprise architecture assignments - both EAI and IT Infrastructure - and project management of system integration projects. Gerrit currently focuses on Identity Management and Directory Services.

Gerrit can be contacted at gerrit@navit.co.za.

Evan Erwee

Evan has a vast experience for 14 years in IT projects as an architect, consultant and engineer. He was co-founder of InfraSoft, a successful infrastructure and software company in South Africa in 1996. Evan is specialised in Directory Services and Identity Management. He is since 2003 a member of Microsoft's Most Valuable Professionals (MVP) program for Directory Services.

Evan can be contacted at evan@erwee.com.

ACKNOWLEDGEMENT

The authors would like to thank Andreas Luther (MIIS Product Manager) for his valuable contribution to this white paper.