Carrier-grade J2EE: The foundation of the Oracle SDP

An Oracle White Paper
April 2006
TABLE OF CONTENT

Executive Summary ........................................................................................................ 3
Introduction .................................................................................................................. 4
  The service delivery challenge.................................................................................. 5
  Relevant Oracle products today............................................................................... 6
The Oracle SDP Vision.................................................................................................. 7
  Carrier grade execution environment .................................................................... 8
  Enablers and Enabler framework .......................................................................... 8
  SOA, Service gateway and Unified user profile.................................................... 9
Out of the box services............................................................................................... 10
Looking back at the web – SDP and the Enterprise................................................ 11
Carrier-grade J2EE – what’s the Fuss? ................................................................. 12
  Availability.............................................................................................................. 12
  Throughput.............................................................................................................. 13
  Latency ................................................................................................................... 13
  Management .......................................................................................................... 13
  EDA......................................................................................................................... 14
Oracle SDP – J2EE realization of the OSE............................................................ 14
  OSE – The OMA Service environment ............................................................... 14
  Parlay and Parlay X............................................................................................... 15
Conclusions............................................................................................................... 16
EXECUTIVE SUMMARY

As IT and Network infrastructure starts to converge within operators and service providers, there is the need for a standards-based environment for providing converged data services on today’s and tomorrow’s networks. Such an environment is called the Service Delivery Platform.

The technology foundation for Oracle’s Service Delivery Platform is the widely adopted industry standard Java EE (formerly known as J2EE) enhanced to be carrier-grade. As a result, a service provider that deploys the Oracle Service Delivery platform can depend on a widely adopted industry standard (over 3 million developers are familiar with J2EE concepts) for the development of exciting, new data services. At the same time, these services can be deployed to have the same level of availability, reliability, stability and performance that operators provide as a matter of course to their customers.

Given the complexity of a general service delivery platform, it is highly desirable for a service provider or an operator to have a standard architecture blueprint for the various components of an SDP. The OMA Service Environment (OSE), a standards effort within the Open Mobile Alliance (OMA) provides such a blueprint. Oracle has been a key driver of the OSE standardization work at OMA. The Oracle Service Delivery Platform is a realization of the OSE on a carrier-grade J2EE environment.

The Oracle SDP is targeted at mobile, voice, multimedia and communication services (e.g., VoIP). It is aimed at telecom operators and service providers; for wireless, fixed, broadband and converged networks as well as to Internet service providers and enterprises that wish to provide next generation data services to their employees and customers. This white paper details Oracle’s vision and the concepts behind a J2EE-based open standard realization of the OSE for service providers as well as enterprises.
INTRODUCTION

Telco and IT technologies are rapidly converging. Confronted with huge infrastructure costs, Telecommunication service providers need to develop new high value services faster to recoup their investments. At the same time, they have to deal with eroding revenue streams from traditional services (e.g., voice) and fend off new aggressive competitors such as virtual service providers and other internet-based service providers that provide competitive services at very low costs. For service providers, it is essential to be able to rationalize their environment to build, deploy and manage services as well as facilitate the development of new attractive services by third party partner service providers.

An example of a new and exciting service is triple play (voice + data + video) combined with integrated billing from a single operator or a service provider to a consumer. The demand for such converged services, combined with the convergence between wireless and fixed networks (e.g., broadband) further emphasizes the need to move away from silo-based architectures and instead rely on a horizontal, standards-based platform that enables convergence – enter a standards-based Service Delivery Platform (SDP).

The telecommunications industry realizes the need for an SDP, particularly one that conforms to industry standards and is built on a standard architecture blueprint. The Open Mobile Alliance (OMA), a federation of industry leaders, is in the process of standardizing an architecture blueprint for SDPs under the OMA Service Environment (OSE) effort. Oracle is a leader in this standardization effort and the Oracle Service Delivery Platform is a realization of the OSE based on Java EE (formerly known as J2EE).

J2EE has been enormously successful as a platform technology that has fundamentally changed the way Internet-based applications are developed. Indeed, its success is easily demonstrated by the fact that over three million developers are familiar with the concepts of J2EE and are able to build new, exciting applications based on J2EE quickly and rapidly. Over the last 10 years, J2EE has also matured as a technology platform to provide the robustness, scalability, availability and performance that are so central to the value proposition of an operator-provided service. Oracle Fusion Middleware is an industry-leading J2EE-based middleware platform that solves the problem of efficient development, deployment and customization of highly scalable and performant applications.

This white paper provides an overview of the Oracle Service Delivery Platform vision and the role of J2EE in the realization of this vision. The companion white papers (Oracle SDP – Enablers and Future-proofing your Investment and Oracle SDP – Profile based policy enforcement) provide additional details on the implementation of key concepts. The rest of this document is organized as follows: we first provide an overview of the key convergence trends and draw parallels to the evolution of the Internet/web. We next provide the vision of the Oracle SDP, and highlight the key components. We next discuss the key attributes
of carrier-grade J2EE. Finally, we place the Oracle SDP in the context of the OSE effort.

**The service delivery challenge**

Convergence is taking place across telecommunication networks at multiple levels. At the transport level, IP based technology is replacing traditional circuit-switching. At the services level, new offerings such as triple play are converging traditional services (such as voice) with new services such as video and data. The consolidation of the industry is leading to a single owner for networks that have been historically disparate (e.g., fixed and mobile networks that are converging under a single owner). Finally, billing for converged services is also converging, with bundling of services and simplification of end-user billing. As a result of all these trends, operators are looking to move from traditional, highly expensive and specialized equipment coupled with closed and proprietary environments to internet-like networks and infrastructure based on commodity hardware and standard software.

To add to the pressure, many operators are facing huge network infrastructure upgrade bills to enable new generation data services. Increasingly, operators will need to provide an attractive package of innovative data services to minimize churn and to keep their existing customers bases loyal. The adoption of services trails the investments by a significant amount (months in the best case to as much as 3-5 years), and since there is no killer application, operators need a flexible environment where they can trial new services quickly, obtain customer feedback and deploy those services that have market traction. This is quite different from the traditional telecom business and deployment model, where there were a number of simple, well defined services, competition was often regulated and operators could afford a fast-follower behavior for new services such as Push-to-talk over cellular (PoC). In short, the operators face the problem of “doing more with less” in an era of increasing demands from customers and increased need for flexibility. This is the heart of the service delivery challenge.

Interestingly, this situation is analogous to the commercialization of the web, where IT service providers and enterprises have had to rapidly build new applications with web based user interfaces and connect the new applications to their backend systems, quickly and flexibly. In almost every instance, this has required the breaking down of traditional silos of application development. For example, inventory management systems had been traditionally developed with no thought of being potentially used by millions of users external to the enterprise firewall. This situation changed dramatically when the Web came along – all of a sudden, anyone who ordered a computer online would want to check the status of their order prior to order shipment.

More recently, the second wave of the web (Web 2.0) has led to architectures that efficiently solve the problem of application integration with other third party services. Technologies such as XML and web services combined with the principles of Service Oriented architectures (SOA) and products such as the Enterprise
Service Bus (ESB) are enabling enterprises to solve the problem of “doing more with less”, and unlock the capabilities of their IT systems for Web-based access by their customers, partners and employees. On the vendor side, Oracle in particular has led the charge by providing the necessary products and technologies as part of Oracle Fusion Middleware.

The centerpiece of the Oracle Service Delivery Platform strategy is that with a number of well defined extensions to the Oracle Fusion Middleware and other Oracle products, the service delivery problem for operators and service providers can be solved quite effectively. Carrier-grade J2EE is at the heart of these extensions.

**Relevant Oracle products today**

Oracle Fusion Middleware provides a standards-based, high performance middleware platform for SOA based on J2EE. High availability, reliability and grid computing initiatives well underway enable Oracle Fusion Middleware today to satisfy requirements that Operators will require for service delivery. (e.g. five 9s of uptime). The Oracle Applications, including PeopleSoft, JD Edwards and Siebel provide OSS and BSS applications that complement Oracle Fusion Middleware by enabling out-of-the-box OSS/BSS integration. Oracle's acquisition of Portal Software adds Billing and Revenue Management solutions relevant to the Communications and Media industries to Oracle’s BSS Portfolio. The combination of Oracle Financials, Oracle inventory Management, Siebel Order Management and Fulfillment, Siebel CRM, Siebel Analytics, and Portal Software constitute the most complete BSS suite for telecommunication providers. Oracle data hubs permit the consolidation of multiple sources of data (e.g., subscriber profile) into a single, consistent and secure view for master identities such as “customer”. When combined with Oracle Identity management, all of the above enable operators and service providers to consolidate OSS and BSS across services and network technologies as well as mix and match new and legacy systems.

Oracle’s database and Real Application Cluster (Oracle RAC) products are widely regarded as industry leaders in terms of performance, reliability, availability and scalability, and are the de facto industry standard in telecom operators and service providers. Oracle TimesTen provides fast in-memory access to data stored in a relational database. The Oracle Carrier Grade Framework (CGF) initiative leverages Oracle RAC and Oracle TimesTen as a base foundation for Telecom Equipment Manufactures (TEM), Independent Software Vendors (ISV), and Communication Service Providers (CSP) to use when building solutions such as a Home Location Register (HLR)/Home Subscriber System (HSS) or a Provisioning System. Oracle TimesTen provides the real-time data management layer and Oracle RAC is the highly distributable persistency layer. The two layers are kept synchronized using the Cache Connect to Oracle feature from TimesTen.

The Oracle Service Delivery Platform will build on this rich and comprehensive base and will selectively enhance the capabilities of the existing Oracle products and
add new products to solve the service delivery challenge. The Oracle SDP is
targeted at enabling the development and deployment of a new generation of
consumer services as well as new services aimed at tapping the market of enterprise
mobilization and enterprise communications. The market for these new generation
services is still evolving and the goal of the Oracle SDP is to help operators,
enterprises and service providers to tap into this rapidly growing market.

THE ORACLE SDP VISION

At its simplest, the Oracle Service Delivery Platform is a standards-based
environment for the design, development, deployment, and management of
services that lie at the heart of IT and network convergence. Oracle SDP is a
natural extension of Oracle Fusion Middleware for telecommunications service
delivery. In particular, Oracle SDP has the following features:

- **A carrier-grade execution environment** for services that access next-
geneneration as well as traditional networks. The explicit design of the SDP
that enables services to work on today’s networks as well as tomorrows
ensures future proofing. This is particularly relevant from an ROI
perspective, since next generation services must be deployed on today’s
networks for the next 12 to 18 months as new networks (such as IMS)
come into being.

- **Enablers and Enabler framework** that provide development time and
run-time encapsulation of key components and building blocks that are
critical to service development.

- Components that bring **SOA** (Service Oriented Architecture) principles to
a service delivery context by enabling orchestration, policy enforcement
and composition.

- **A service gateway** that allows partners controlled access to critical
network resources with SLA enforcements.

- **A unified user profile** that provides a unified view of user (subscriber)
information to the service layer and execution environment in the
presence of silos of information and multiple networks

- **Out of the box services** built on the standards-based platform that can
be used to demonstrate immediate ROI.

The SDP opens up the network to a vast community of services and developers
without sacrificing the management, security and control necessary to deliver the
quality of service the telecom industry demands. We discuss each of the SDP
features next.

**Carrier grade execution environment**

Any execution environment that is at the heart of the service delivery for operators
and service providers must be carrier grade. We discuss the carrier grade
requirements in detail in the section of the suitability of J2EE for this purpose – here we provide a quick overview of the key requirements:

- A carrier grade execution environment must be **highly available**
- Such an environment must be **highly scalable** and support a large number of transactions in a cost effective manner – this has implications for both horizontal and vertical scalability
- Such an environment must be suitable for the most demanding applications from a **latency requirement** perspective (e.g., call control). In the PSTN world, the general expectation is that within 500 milliseconds of the last number being dialed, the calling party can hear the ring tone from the other side – a carrier-grade execution environment must be able to satisfy this scenario
- A carrier-grade execution environment must support the easy deployment, **management** and upgrade of services
- Telecom services are fundamentally different from web-based applications in that they are event driven. A carrier-grade execution environment must efficiently support an event-driven application programming model (EDA).

**Enablers and Enabler framework**

Enablers are building blocks that expose a simple northbound API (with a Java or a web service binding for example), and isolate the user from the complexities of the underlying network capabilities. Enablers are a key pillar to the overall strategy of future-proofing service development and expanding the reach of developers that are able and willing to develop network-centric applications. In the Oracle SDP, enablers will typically provide the following features and benefits:

- **Abstraction** of underlying network technology choices and settings. For example, regardless of the vendor, the specific protocol extension used by a specific vendor, an enabler can expose the same capability “northbound” to an application developer. This can guarantee the stabilization of application development APIs with respect to underlying network technology choices. As a result, migration at the network level and integration with existing and future network elements can be carried out incrementally without wholesale replacement of silos as is the case today.
- **API-based** support for service integration. By supporting a broad range of APIs (Java or C-based vs Web services based), enablers can permit a number of integration scenarios including in-house development as well as integration with 3rd parties. This improves operator and service provider flexibility when they make a decision on whether the implementation of a particular feature should be done by their own developers or whether such an effort can be outsourced to 3rd parties. Also, API-based application integration offers
significant benefits when compared to the traditional Telecom approach of protocol-based component integration.

- Simpler OSS/BSS integration. By providing enablers (such as charging) that encapsulate common OSS/BSS integration requirements, operators can consolidate and share OSS/BSS systems across multiple services. Consolidation across multiple networks and network technologies is critical to support convergence between wireless, wired and broadband networks, for example. Having a unified view of charging, identity management, subscriber profile, CRM and PRM systems makes service migration and network technologies and vendors significantly more cost effective when compared to a silo-based approach.

In addition to providing out-of-the-box enablers that provide common building blocks, the Oracle SDP will also provide an enabler framework that will allow ISVs, SIs, operators and service providers to develop their own enablers. This ability to customize and extend the base capabilities provided by the SDP using IT standard technologies is at the heart of the value proposition for the SDP.

The Oracle Service Delivery Platform will include several out of the box enablers that have the characteristics defined above and which expose common, reusable functionality:

- SIP Servlet container
- Call control
- Presence
- Media server control
- Messaging

Additional details on the Enablers listed above and the Enabler Framework that is provided as part of the Oracle SDP are available in the companion white paper Oracle SDP – Enablers and future proofing your investment.

**SOA, Service gateway and Unified user profile**

The core principles of SOA – reuse, delegation, composition and service orchestration – are as applicable to services developed in a Telecom domain as they are in the Enterprise/IT/Internet domain. For example, an operator may wish to compose a bundle of services differently in different geographic areas because of competitive reasons or to target different customer segments. At the same time, operations such as orchestration are required not to degrade the performance of the core services and applications significantly. Services need to be exposed to 3rd parties in a way that is secure, manageable and billable. Services should be able to support delegation of certain functions – for example, before a service can be used by any user, there may be a common authentication step that needs to be performed. A service invocation must be able to accomplish this user
authentication by delegating that function to a common authentication capability and based on the result of the authentication step, proceed with the initial service invocation as appropriate.

Services should also be able to support the invocation of business rules and policies during their invocation. These business rules and policies generally come in three different flavors: (a) pre-service invocation for authentication and authorization, (b) service-specific policies that require user-level, subscriber-level or service-level information, and (c) network-element specific policies that govern how the service may access the underlying network resources. For example, a premium messaging service may have any combination of the following three policies applied during its invocation by a user:

- First, a user is authenticated using their user ID and password information, after which the user is also authorized as someone having the ability to use the premium messaging service
- During the execution of the premium messaging service itself, the service may make use of a white list or a black list to determine whether this user is allowed to send a premium message to other 3rd parties
- Finally, the premium messages sent by a user might be directed to a specific set of underlying network elements (SMSC or MMSC) based on the user profile as well as the capacity in the network. For example, during the execution of a time-critical application such as Tele-voting, it may be acceptable to delay the delivery of any other premium content that is not time-critical.

Finally, self service tools for 3rd parties, subscribers as well as service administrators are a key to SOA-enablement of Telecom services.

In the scenario outlined above, an important consideration is the separation of the service logic from the policies that can be enforced during service execution. This separation serves two purposes: First of all, the resulting flexibility in service behavior results in a service (or an enabler) being usable in multiple scenarios, with no need for developer intervention or changes to the underlying implementation – a core SOA principle. An added benefit is that a highly declarative framework (e.g., BPEL or Rulesets) can be used to specify the policies that should be enforced during service execution, enabling business owners, end-users and administrators of a service to modify its behavior dynamically.

The unified user profile is a single, virtual repository that provides a service developer with a unified view of all subscriber and user information that is required for efficient service execution in an SOA context. The focus in the unified user profile is on static profile data – for example, subscriber identity, services used, MSISDN, roaming network information etc. The user profile has to support a set of standard interfaces northbound for access by a service developer (e.g., through Diameter), and must be able to aggregate information from a variety of underlying
data sources, including the provisioning systems, the HSS etc. Such a system must scale to support tens of millions of users, while at the same time supporting fast in-memory access to critical service data that is required for evaluating business policies on service invocation. The high-performance integration between the subscriber profile and the enforcement of service-specific policies and service orchestration is central to personalizing the service behavior for a subscriber. This integration is a key differentiator for the Oracle Service Delivery Platform strategy. More details on the interplay between the Oracle SDP components that enable SOA, the Service gateway and the unified user profile are available in the companion white paper *Oracle SDP – Profile based policy enforcement.*

**Out of the box services**

Finally, Oracle recognizes that compelling as a service delivery platform might be, the real value of the platform approach is when the 4th or the 5th or the 6th service is deployed by a service provider. In particular, when the first 2 or 3 services are being deployed, the business case for a platform is quite hard to make, since the first 2 or 3 services typically have to provide the business justification for themselves without the additional burden justifying the necessity for a platform. As a result, Oracle will provide a number of out of the box services built on top of the standards-based platform that will provide a choice to an operator for an immediate business case for the SDP deployment. The services currently being planned are the following:

- **Virtual PBX:** Providing a PBX service that works across existing networks and NGN; and across fixed and mobile networks is particularly compelling as a hosted offering to the enterprise.
- **E-mail push:** Providing the Blackberry user experience on any device from any email back-end using the open IETF standard P-IMAP is particularly compelling to the mobile service providers
- **Content delivery:** Providing the end-to-end management of the delivery of rich content, all the way from content partner on-boarding to the download of the rich content to the device while maintaining the content rights for the content provider is a challenge every 3G operator faces.
- **Residential VOIP:** Providing a first line telephony replacement service that provides the traditional residential voice features as well as next generation features such as web-based call management. Regulatory requirements (such as anonymous call rejection) and legal requirements (such as CALEA) are provided as well.

In addition, Oracle is committed to working with the ISV community to foster the development of new applications and services and the porting of existing services and applications to the Oracle Service Delivery Platform. The combination of Oracle’s commitment to a strong standard based service platform and the specialized domain expertise that ISVs bring to the table will benefit Oracle customers by providing a broad base of revenue-generating applications on a standards-based platform.

**Looking back at the web – SDP and the Enterprise**

The challenges faced by the Telco service providers while solving the problem of service delivery are eerily similar to those faced by enterprises when dealing with
the web and the Internet. Solving these issues back then resulted in the introduction of the application server and J2EE, and was further refined through the concepts of Service-Oriented Architecture (SOA) as an open standard-based environment. Today such a platform has evolved into a full-fledge application development, deployment and management platform with support for business integration and automation of 3rd party management.

In particular, all of the following SOA principles that are today embodied in the Oracle Fusion Middleware platform are also guiding principles for the Oracle Service Delivery Platform:

- Loosely coupled, interoperable and reusable application components
- Layered architecture for capability integration and consolidation
- Standard interfaces
- Event-driven, message oriented architecture that supports ease of integration
- Ease of orchestration for improved usage via workflow and business process definition
- Policy enforcement for securing service usage
- Local and distributed execution alternatives through the use of J2EE or web services
- Business Intelligence and Business Activity Monitoring for real time and historical insight into business performance

Given the philosophical heritage of the Oracle SDP, it should not be a surprise that the Oracle SDP can also solve the enterprise problems around integration of the communication and computing infrastructure. For example, mobilizing applications, being able to integrate a VoIP capability to an existing enterprise portal is a simple use case for the call control enabler. Enterprises have different pricing and packaging requirements than the service providers and the operators, but the core technology required to tightly integrate communications into enterprise computing is indeed part of the Oracle Service Delivery Platform product set.

**CARRIER-GRADE J2EE – WHAT’S THE FUSS?**

Over the last few years the J2EE platform has emerged as the popular choice for building highly scalable transactional systems across a variety of industries using Java in the middle-tier. The Oracle J2EE platform has been optimized for transaction throughput, scalability and reliability through countless deployments in mission critical applications. J2EE based-middleware is at the heart of the Oracle SDP. In this section, we examine the key requirements for an infrastructure to be carrier grade and we examine how/if those requirements can be achieved using J2EE.

**Availability**

In a telecom or service provider environment, services need to be highly available. J2EE has matured as a technology to provide the level of availability required for
the most mission critical Telecom services. Indeed, many mission critical systems in a non-Telecom environment already are implemented today in a J2EE environment – for example, the entire air traffic control for the Western United States is implemented on top of a J2EE based application server.

Leveraging its J2EE core, the Oracle SDP will provide the following capabilities to address the availability requirements on a telecom service:

- Local (or LAN-based) availability: J2EE uses clustering to handle the failure of a single node that may be handling the delivery of a telecom service to an end user. Historically, high availability offerings have focused on achieving high node-level availability through hardware redundancy and dual-system designs. J2EE takes a cluster based approach where a single node in a cluster may not have 5 9's availability (in fact, it does not), but the cluster, and therefore, the service as a whole can in fact have the level of availability desired from a telecom service.

- WAN (or geographic) availability: By distributing a cluster over a widely dispersed geographic area, it is possible to guard against catastrophic (e.g., data center) failure.

- Rolling service upgrade: Finally, it is possible, in a J2EE-cluster, to upgrade the deployment of a service in a cluster node by node without service downtime. A very common approach is to use a load balancer configuration so that no new requests are routed to any given node, after which that node is “drained” of its existing service obligations. Once a node has been taken out of commission, the service is upgraded on that node and the node is restarted with the new version of the service. The load balancer configuration is updated so that new traffic from new users is directed to nodes with the upgraded service instance and as this process “rolls” through the entire cluster, it is possible to upgrade the entire service with no downtime. It is also possible to make this process transactional so that if the upgrade does not complete successfully, the cluster configuration can be rolled back to the earlier configuration so that there is no service downtime.

**Throughput**

Throughput has been a historical strength of J2EE-based application servers. For example, the e-commerce not-for-profit transaction site, Comic Relief, hosts the Red Nose Day as a biannual, nationwide fundraising event for reducing world poverty. The most recent Red Nose Day resulted in an unprecedented audience (exceeding 13 million), which in turn led to the comic Relief e-commerce web site handling more than 700 transactions per second on 100+ CPUs of Oracle Fusion Middleware. Other high-throughput transaction sites deployed today on Oracle Fusion Middleware include monster.com – the world’s top job search web site, Telecheck – a leading provider of paper and electronic check services that handles 3.6 billion transactions annually and Posco – the world’s largest J2EE deployment.
As part of the Oracle SDP roll-out through 2006, Oracle is conducting a number of benchmarks (including for call control) that should demonstrate conclusively that the throughput strengths of J2EE in the Internet domain also can carry over to the service provider and Telecom domains.

**Latency**

Low latency operation is a key requirement of Telco deployments. There are three main technology areas that contributes to make the Oracle SDP platform a low latency environment:

- **Virtual Machine Optimization**: in the recent years Virtual machine technologies (HotSpot, JRockit etc) has matured to a point whereby it is possible to guarantee essentially deterministic pause times using a standard garbage collector for most J2EE container operations.

- **Enterprise Service Bus**: In addition, the enterprise service bus (ESB) technology can be used to complement garbage collector performance improvements to manage overall application latency. The ESB enables low latency data transfer operations (with guaranteed maximums on delays), is distributed and highly scalable. By providing a real-time, message-oriented backbone, the ESB can help manage the latency for an application operation that crosses the boundary of a single processing element.

**Management**

Services deployed in a telecom environment have unique management requirements, often because of regulatory reasons. For example, methods and procedures need to be in place to define the actions that should be undertaken in the event of a failure of a node in a cluster that is running an instance of a telecom service. These methods and procedures have to be defined before the service ever goes into production and must represent many possible contingencies that relate to service failure.

As a prerequisite to providing standard methods and procedures for various failure scenarios, it is also required for the SDP (or the service execution environment) to integrate with various management consoles such as Tivoli, BMC Patrol or MicroMuse through the generation of SNMP traps. SNMP traps can be generated for informational purposes (e.g., when a certain number of service requests have been satisfied), notification purposes (e.g., when certain critical system thresholds have been reached – e.g., memory usage) or emergency scenarios (e.g., node failure in a cluster). The ability to define custom MIBs to extend the out of the box SNMP traps that an SDP vendor may provide is also essential.

Finally, the ability to provide command line interfaces to perform the routine actions indicated in the methods and procedures is also required. For example, when a node fails, it may be required to manually take it out of a cluster, re-start the node, and bring it back into cluster membership. It should be possible to
accomplish these goals through standard, well-documented command line interfaces.

**Event Drive Architecture**

Finally, an execution platform for telecom services needs to efficiently implement an event-based programming model. Telecommunication services are inherently asynchronous and event-driven. The network generates various events that must be efficiently processed by the application infrastructure (the SDP in our case), and an application developer must be able to easily write an application that is primarily event driven. Such an application must also execute efficiently in the SDP environment.

A J2EE-based application server that supports JCA 1.5 coupled with the enterprise service bus (ESB) handles this requirement effectively. JCA is a bidirectional adapter framework that allows an application to both send notifications to the underlying network as well process events generated by the network element. The enterprise service bus through its message oriented, asynchronous publish/subscribe based model is able to deliver events to their intended destination efficiently. Together, these implement an event-driven architecture that supports the telecom requirements for an event processing system. Such a system is able to provide acceptable performance from a telecom event processing perspective (e.g., guaranteed latencies for example) and match a SLEE environment in this regard.

**ORACLE SDP – J2EE REALIZATION OF THE OSE**

The Oracle vision for SDP, Oracle’s SDP realization and Oracle’s efforts to standardize SDP within the industry (e.g. OSE) are all based on the concepts of SOA and its expertise in developing IT middleware, tools and applications. Oracle has been a leader in defining the Open Service Environment in the OMA and in driving the evangelization of this effort in the marketplace.

**OSE – The OMA Service environment**

The OMA Service Environment (OSE) has been approved by the OMA as its architecture model that describes interaction to, from and among OMA enablers. Oracle has been involved from the beginning in driving some of its key concepts. The OSE conceptual model is illustrated in Figure 2.

The OSE was introduced by OMA to address the problems of most Telco technologies that are based on vertical silos with little re-use and compatibility across enablers. OSE evangelization is still work in progress at OMA.

Today there is still no full consensus in Telecommunications industry of how the OSE can model new horizontal platforms demanded by Service Providers. Oracle has taken the position that the Oracle SDP shall be a carrier-grade J2EE based realization of the OSE. This direction is based on existing implementations of SDP-like functionality at multiple Oracle customers who have validated this approach in real life, particularly with mobile operators.
Parlay and Parlay X

Parlay and Parlay X have been defined by the Parlay group in collaboration with other SDOs (Standard Development Organizations). In our vision, Parlay integrates smoothly with the OSE and Oracle SDP at two levels:

- A Parlay framework and gateway that rationalizes Parlay-compliant underlying servers.
- Parlay X components that provide reusable Web Service (WS) modules easily exposed to remote application developers.

Parlay provides access through its APIs to PSTN call control and Intelligent Network (IN) resources. Parlay (known as Parlay-OSA) provides the recommended way to develop services over the IMS (IP multimedia Sub-system) as specified by 3GPP and MDS (Multimedia Delivery Service) the corresponding system specified by 3GPP2. Parlay provides standard OSE / SDP drivers and resources that can be driven by these drivers. Parlay X enabled components are functionally equivalent to OMA enablers except their binding is already selected to be WS. Oracle’s view of the relationship between OSE and Parlay is illustrated in Figure 3.

An SDP based on the OSE is future proof: once services and enablers are abstracted from the underlying network. In today’s networks, Parlay and proprietary drivers can be used to provide PSTN and IN services. As networks evolve from 2G to 3G and start to deploy IMS, transition initially does not even require changes of drivers. However, smooth transition to Parlay and IMS is also ensured by using directly SIP and IMS APIs as drivers. As a result, the same platform and applications to be used (simultaneously) over wireless, wired and broadband (e.g. IP TV) access.
CONCLUSIONS

Operators and Service Providers are in need of an open standards-based Service Delivery Platform (SDP) to competitively and viably provide next generation data services to their users. Such a SDP is a converged IT-Telecom infrastructure suitable for delivering data services based on IT technologies applied to the service provider domain with carrier grade constraints.

The OMA Service Environment (OSE), driven from the beginning by Oracle, provides a blueprint for such an infrastructure. Oracle’s Service Delivery Platform vision is a J2EE realization of OMA’s OSE and has been validated with real life proof points, particularly with mobile operators.

Oracle is the only IT vendor able to offer an end to end SDP with its portfolio of:

- J2EE-based middleware platform that enables SOA out of the box through Oracle Fusion Middleware
- OSS and BSS systems through its application portfolio
- RAC and TimesTen technology that allows the ideal combination of fast real-time access to profile data while separately providing a scalable, persistent data layer
- Out of the box revenue generating services through its portfolio of out of the box services, including those provided by the Oracle Collaboration Suite (OCS).

The Oracle SDP offering is being designed to satisfy the needs of operators and service providers as well as enterprises that need to develop new services or add new features around voice, mobility or communications to existing services. The SDP is an evolutionary platform ideal for development of services that are future-proof and is explicitly designed to handle technology evolutions and today’s convergences (e.g. network technology convergence, network type convergence (wireless, wired and broadband), application convergences (triple play, merging voice, web and multimedia) etc.
Oracle Service Delivery Platform

Figure 1

OMA Service Environment (OSE)

Figure 2

Approved - [http://www.openmobilealliance.org/release_program/ad.html](http://www.openmobilealliance.org/release_program/ad.html)
Parlay, Parlay X, and ParlayInOSE

Parlay GW is just *ONE* way to abstract network resources, direct adapters and SIP are others.