

dcml

Data Center Markup Language

Setting the Standard for Utility Computing

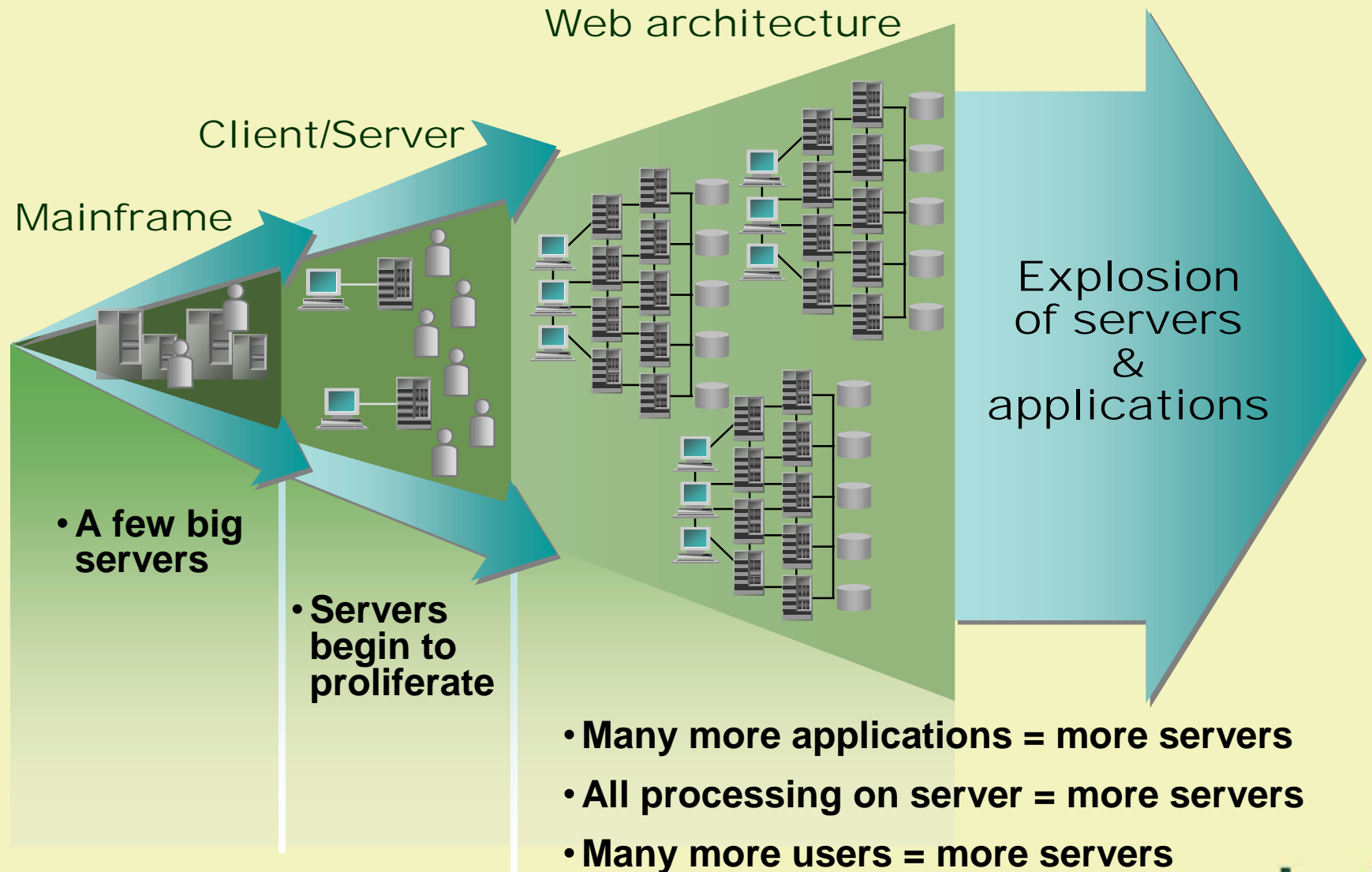
DCML Technical Overview

Tim Howes, CTO, Opsware Inc.

Darrel Thomas, Chief Technologist, EDS
Automated Hosting

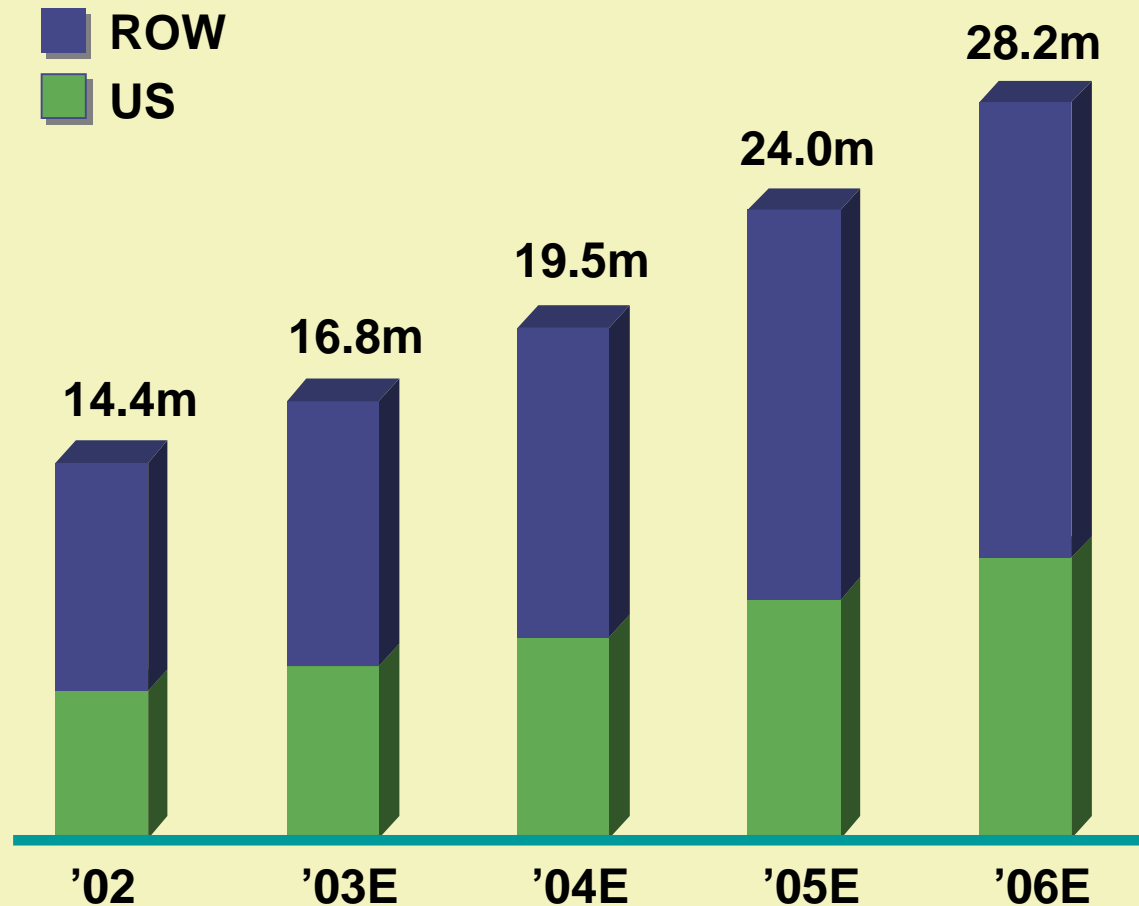
October 14, 2003

Shift to Web Architecture = Explosion of Servers & Applications



No End in Sight to Server Explosion

Worldwide server installed base – Windows, Unix, Linux



Fortune 100 company =
~10,000+ servers today,
~20,000+ servers in 5 years

Fortune 2000 company =
~1,000+ servers today,
~2,000+ servers in 5 years

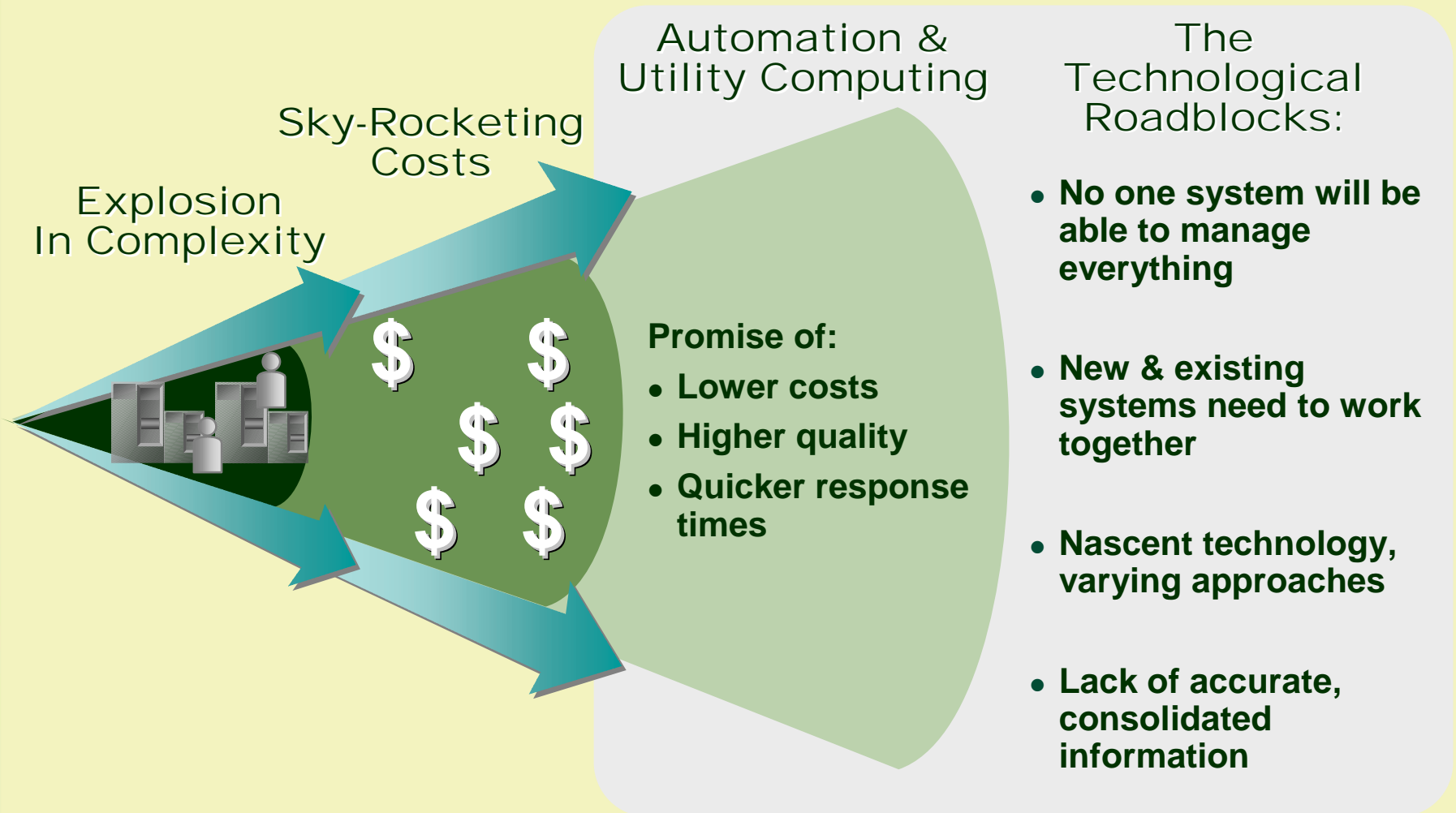
Heavy technology users =
~20,000+ servers today,
~50,000+ servers in 5 years

Source: IDC Server Census, Installed Base

EDS: Trends in Action

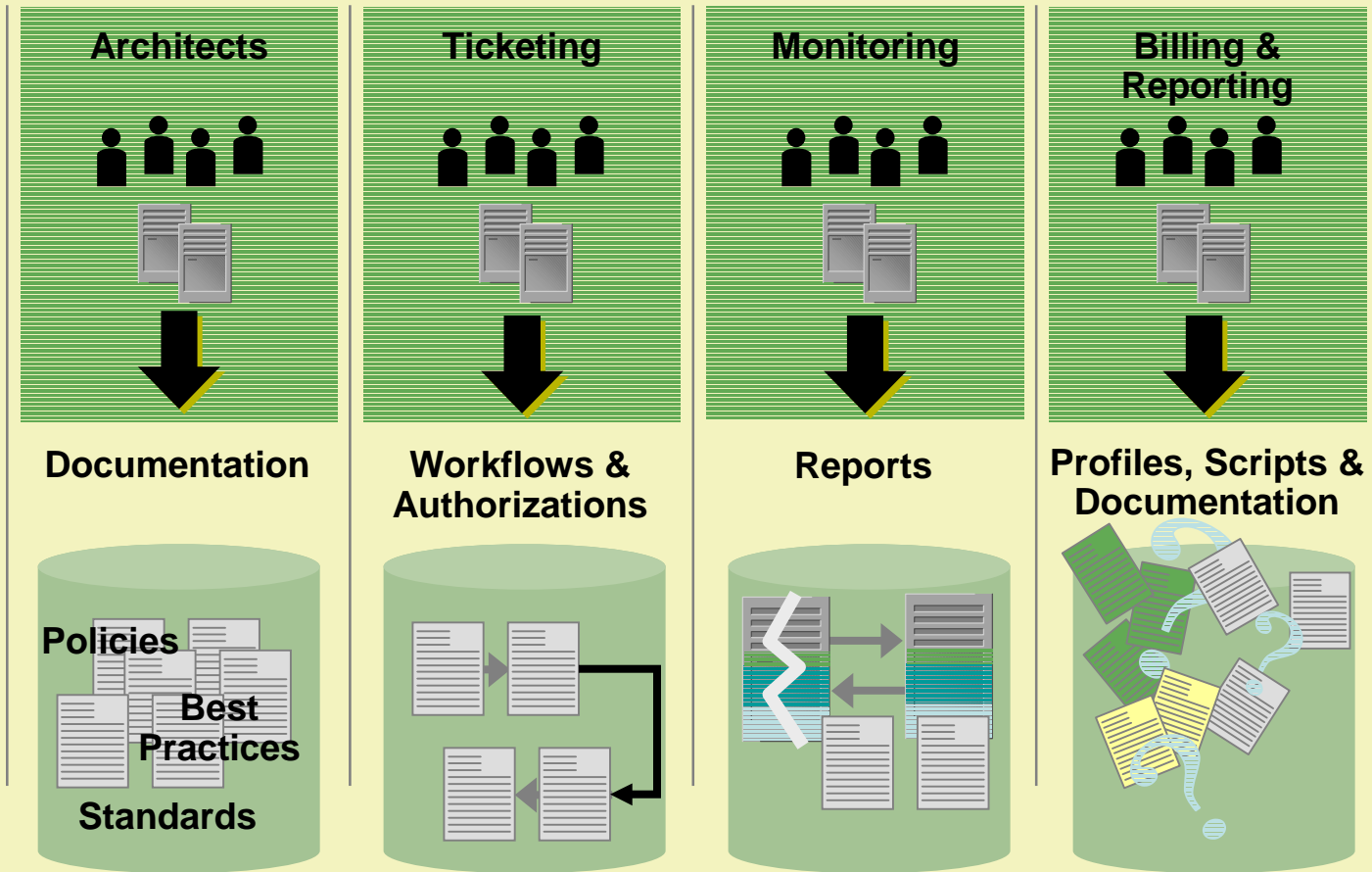
- Growth of Internet-enabled business is 200% year over year
- Number of servers managed has gone from 21,000 in 1997 to over 50,000 in 2003
- Number of packaged applications managed has gone from 10,000 in 1997 to 40,000 in 2003
- Manage over 2.5 billion lines of custom code for over 1 million custom applications
- Rate of change / updates applications has gone from 1 a month to 3 a week
- Percentage of external facing applications has gone from 10% to 80%

Utility Computing: Huge Benefits and Challenges

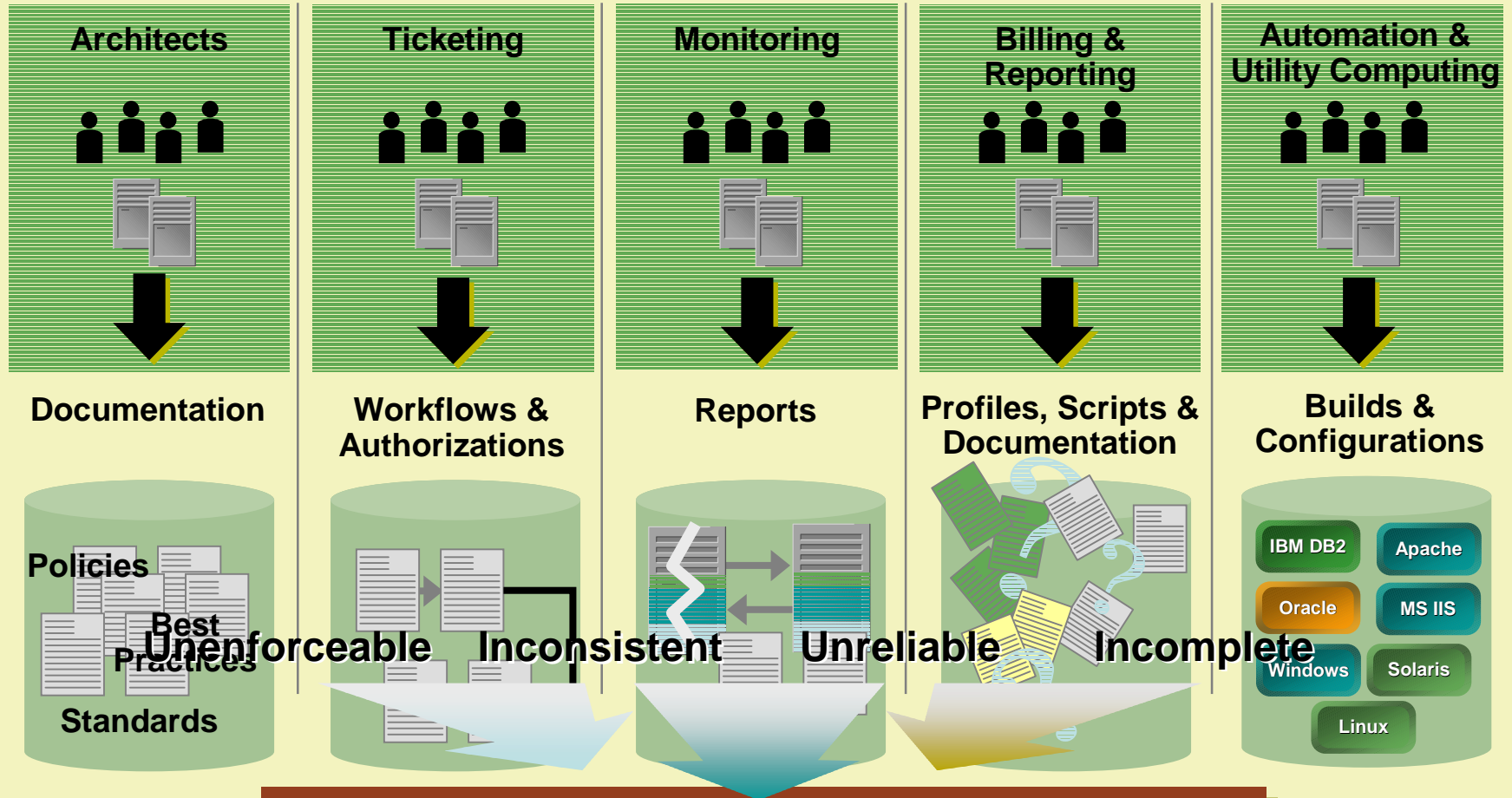


The Challenge: Stovepipe Systems

Different Management Systems, Each with Different Data



The Risk: Utility Computing Becomes Another Stovepipe



Implications:

- Inaccurate, incomplete views of the environment
- Inability to affect change across the entire environment
- Slow response times to changing business needs

Utility Computing Today

Example: Scaling Up an Existing Application

1. Performance management system recognizes slow response time for a critical IT service
2. Monitoring system identifies underlying application component that requires additional resources
3. Provisioning system allocates new resources from pool
4. Automation system reconfigures other resources in the application cluster and reconfigures monitoring
5. Billing system records change for accurate charge back

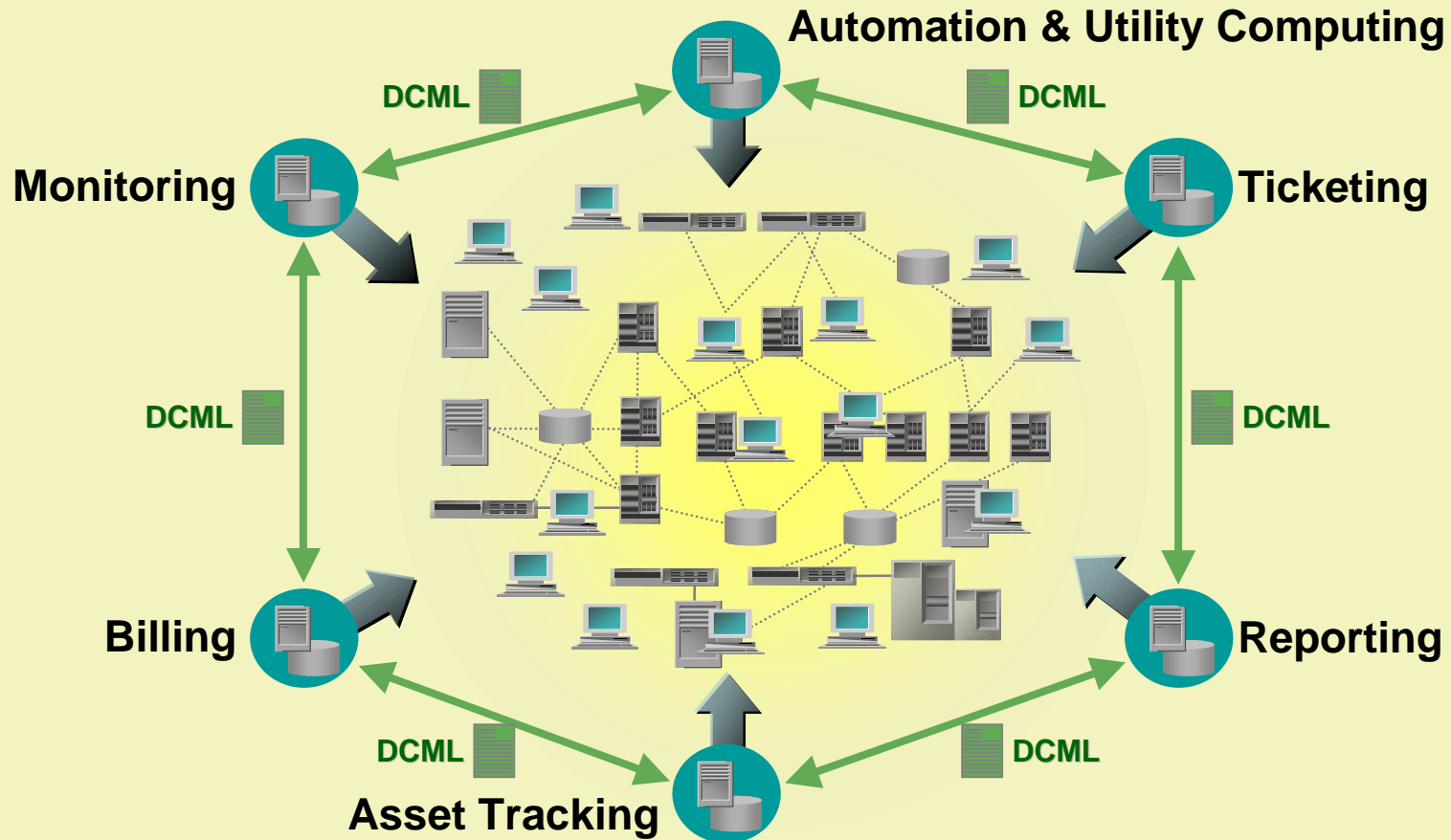
- ▶ At least 5 management systems involved
- ▶ No way to ensure all systems have consistent data
- ▶ Today, typically takes weeks, desired time is minutes or hours

Tremendous manual effort or custom integration is involved

Requirements for Automation & Utility Computing

- A common way to represent and share information among new and existing management systems
- An open specification
- Applicable to heterogeneous environments
- Applicable to existing and new environments

Realizing the Utility Computing Vision



DCML – the first common language that enables automated IT environments

What is DCML?

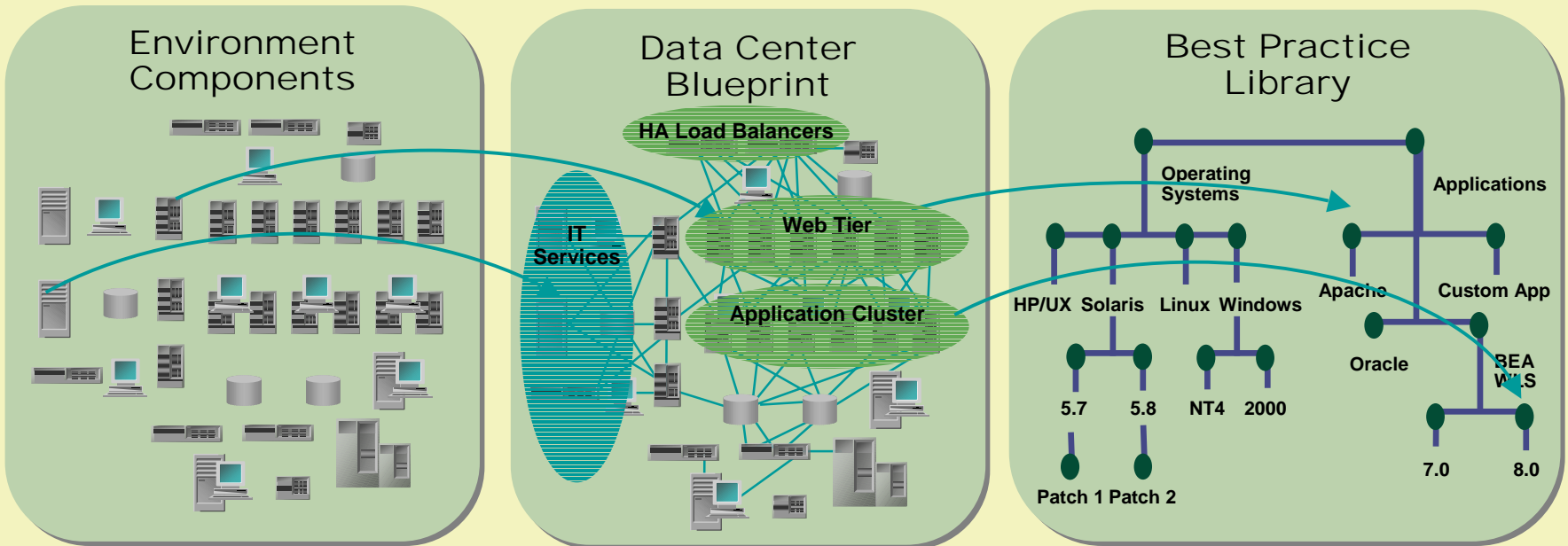
- The Specification
 - Structured XML-based data model and format for describing data center environments, relationships among components, and policies and procedures governing their use and construction
- The Effort
 - Open industry consortium of vendors and customers dedicated to developing the DCML specification and implementing and deploying DCML products
- The Organization
 - Not-for-profit 501(c)(6) organization established to foster development of the specification, submit it to a standards body, and provide ongoing marketing support

What Makes DCML Different?

- ***Provides a recipe, not a catalogue:*** DCML describes how to construct systems under management, not just their state
- ***Goes beyond components:*** DCML also captures best practices, operational policies and standards
- ***Useful to new & legacy management solutions:*** DCML bridges the gaps between new automation systems and existing management systems
- ***Defines a data format:*** DCML specifies the data model and semantics, not protocol or API

The Scope of DCML

DCML



Environment

- Servers
- Network
- Storage
- Facilities
- Customers

Instances

- Services
- Relationships
- Tiers
- Clusters
- Custom configurations
- Constraints

Policies

- Change mgmt windows
- Technology dependencies
- Technology compatibilities
- Standard builds
- Default configurations
- Vendor updates

Example DCML File

DCML name space

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<DCML xmlns="http://www.dcmi.org/ns/2003/dcmi">
  <Header>
    <Creator>
      <System>
        <Vendor>Opsware Inc.</Vendor>
        <System-Opware>DCML Exporter</System-Opware>
        <Version>0.1</Version>
        <VendorURL>http://www.opsware.com/</VendorURL>
        <SystemURL/>
      </System>
    </Creator>
    <Notes>Example DCML document</Notes>
  </Header>
```

Best practice library

```
<Library>
  <Model>
    <OS model_name="NodeOS" is_hidden="false"
        id="OS320009" name="Microsoft Windows 2003" model_id="5">
      <Customers>
        <CustomerRef type="Customer"
                    ref="Customer10007"
                    name="ANY"/>
      </Customers>
      <Platforms>
        <PlatformRef type="Platform"
                    ref="Platform50001"
                    name="Microsoft Windows 2003"/>
      </Platforms>
      <Packages/>
      <Relationships/>
    </OS>
```

Best practice Linux definition

```
<OS model_name="NodeOS" is_hidden="false"
    id="OS420007" name="RedHat Linux AS 2.1" model_id="5">
  <Customers>
    <CustomerRef type="Customer"
                ref="Customer10007"
                name="ANY"/>
  </Customers>
  <Platforms>
    <PlatformRef type="Platform"
                ref="Platform50002"
                name="RedHat Linux AS 2.1"/>
  </Platforms>
  <Packages/>
  <Relationships>
    <ChildOf type="OS" ref="OS5" name="OS"/>
  </Relationships>
</OS>
```

Change management policy

```
<ChangeWindow model_name="ChgWn" is_hidden="false"
id="CW627007" name="Standard Change Management Window">
  <Customers>
    <CustomerRef type="Customer"
                ref="Customer10007"
                name="ANY"/>
  </Customers>
  <Window>
    <WindowStart>00:01</WindowStart>
    <WindowEnd>03:00</WindowEnd>
    <WindowDays>Sun</WindowDays>
    <WindowZone>GMT</WindowZone>
  </Window>
</ChangeWindow>
```

Data center blueprint

```
<Blueprint>
  <Group model_name="Group" is_hidden="false"
        id="Group20007" name="Check Processing Service"
        model_id="13" xmlns="">
    <Customers>
      <CustomerRef type="Customer"
                  ref="Customer30007" name="Finance"/>
    </Customers>
    <Relationships>
      <Contains type="OS" ref="OS420007"
                name="RedHat Linux AS 2.1 Reference Build"/>
      <Contains type="Application"
                ref="Application86590007" name="JBoss 3.2.1 Reference"/>
      <Contains type="Application"
                ref="Application6010007" name="Apache 1.4.7"/>
      <Requires type="SDMReq" ref="SDM840009"
                name="SDM Server Requirement"/>
      <Requires type="SDMReq" ref="SDM840008"
                name="SDM Network Requirement"/>
      <BoundTo type="ChangeWindow" ref="CW627007"
                name="Standard Change Management Window"/>
    </Relationships>
  </Group>
</Blueprint>
```

References to SDM application requirements

Environment components

```
<Environment>
  <Hardware model_name="NodeHardware"
            is_hidden="false" id="Hardware1150007"
            name="m017.london.opsware.com" model_id="3" xmlns="">
    <Customers>
      <CustomerRef type="Customer" ref="Customer30007"
                  name="Finance"/>
    </Customers>
    <LocationRef type="Location" ref="Location1100007"
                 name="London"/>
    <CIMServer>
      <CIMRef 1202758</CIMServer>
    <InstalledSoftwares>
      <InstalledSoftware uninstalleable="true"
                       display_name="JBoss 3.2.1 Reference">
        <Version>3.2.1</Version>
        <Release/>
        <RelatedProduct>JDK 1.4.1</RelatedProduct>
      </InstalledSoftware>
    </InstalledSoftwares>
    <Relationships>
      <MemberOf type="Group" ref="Group20007"
                name="Example"/>
    </Relationships>
  </Hardware>
```

Reference to CIM server definition

Customer definition

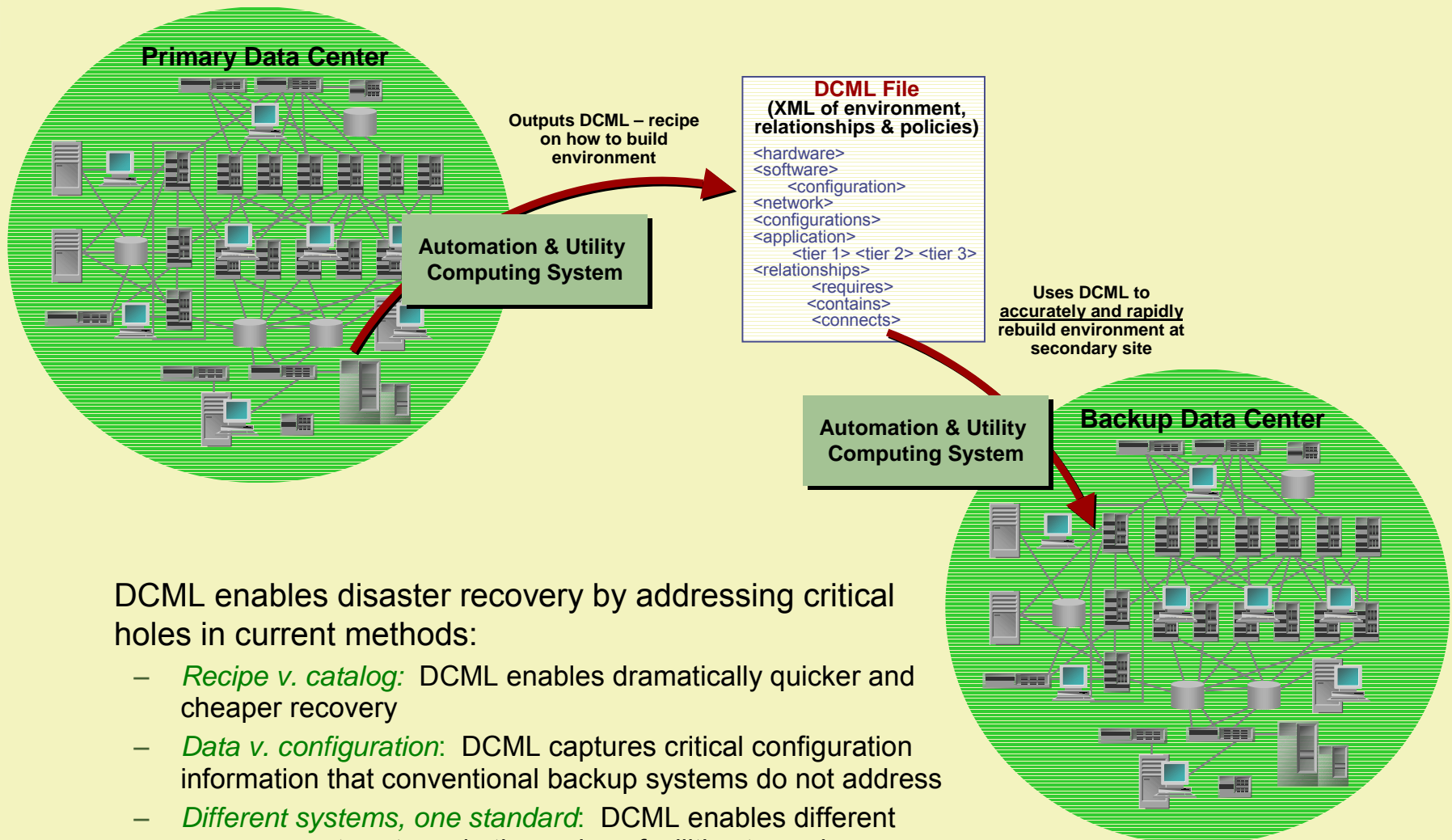
```
<Customer model_name="NodeCustomer"
           is_hidden="false" id="Customer10007"
           name="Finance" model_id="1" xmlns="">
  <Description>Finance Business Unit</Description>
  <CustomerAttributes/>
  <AuthDomain>N/A</AuthDomain>
  <Status>ACTIVE</Status>
  <BusinessCustomerID>-1</BusinessCustomerID>
  <Relationships/>
</Customer>
<Customer model_name="NodeCustomer"
           is_hidden="false" id="Customer30007"
           name="HR" model_id="1" xmlns="">
  <Notes>
    <CustomerAttributes/>
    <AuthDomain>N/A</AuthDomain>
    <Status>ACTIVE</Status>
    <BusinessCustomerID>-1</BusinessCustomerID>
    <Relationships/>
  </Notes>
```

Key Applications of DCML

- **Construction**, when used by automation and utility computing systems or existing IT management systems to reproduce or make changes to systems described in a DCML document
- **Management**, when used to transmit information in a DCML document between automation and utility computing systems and traditional management systems about the state of the managed environment
- **Visibility**, when used to capture and analyze information in a DCML document describing the current state of an environment

DCML in Action

Example 1: DCML Powered Disaster Recovery

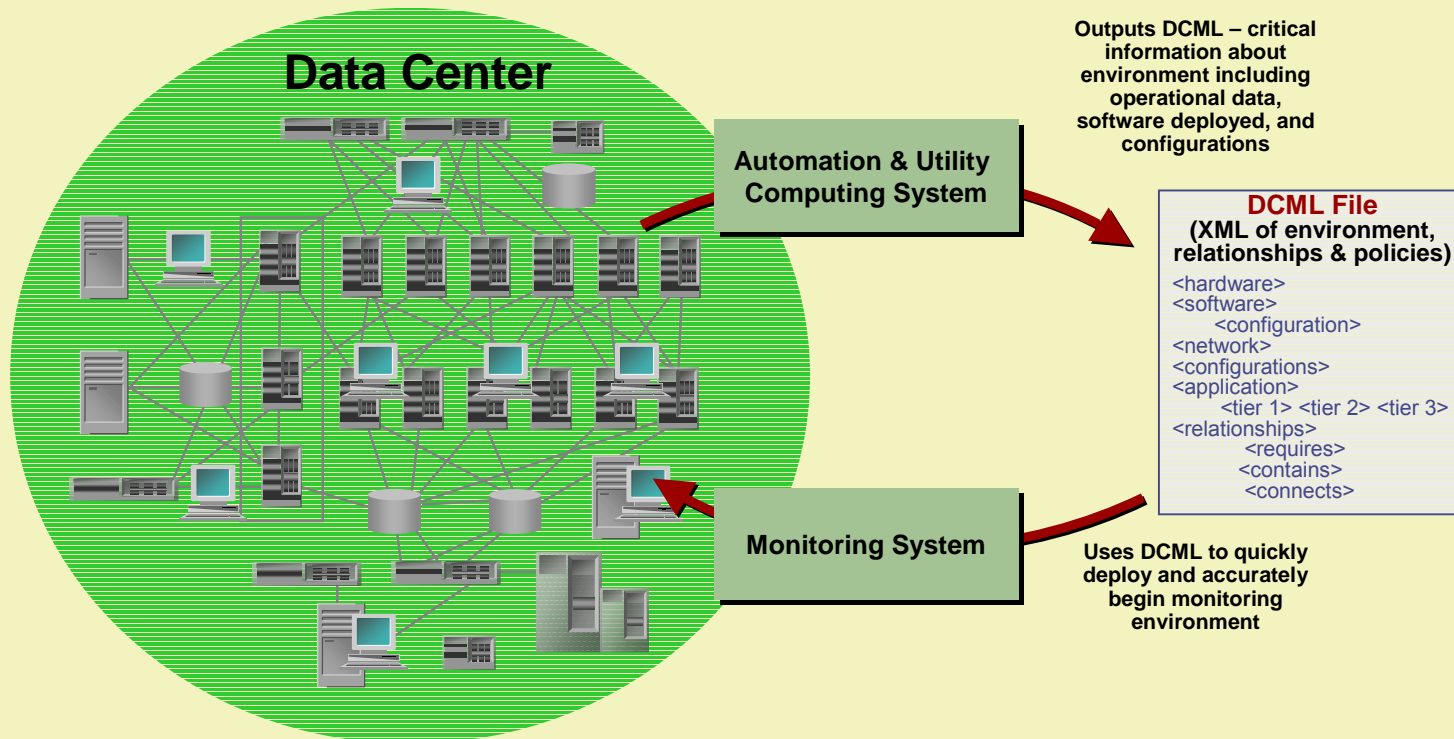


DCML enables disaster recovery by addressing critical holes in current methods:

- *Recipe v. catalog*: DCML enables dramatically quicker and cheaper recovery
- *Data v. configuration*: DCML captures critical configuration information that conventional backup systems do not address
- *Different systems, one standard*: DCML enables different management systems in the various facilities to exchange relevant data

DCML in Action

Example 2: Deploying a New Monitoring System



DCML enables new management systems to be implemented dramatically faster:

- *Enables customers to extract value immediately:* With DCML, new management systems are immediately useful in the environment rather than taking years to successfully implement
- *Going beyond a node list:* By encapsulating critical operational and configuration information, DCML provides the detailed knowledge needed by today's systems
- *Accurate and up to date:* With DCML the latest, highest quality information from many systems can be captured and used

Who Will Drive DCML Forward

Customers with Large Heterogeneous IT Environments

DCML will allow their systems to work together, lowering costs and increasing quality and security

Automation & Utility Computing Vendors

Increased customer adoption of their products and quicker realization of market potential

Enterprise Management System Vendors

Increased value to customers and inroads into automation and utility computing markets

Technology Vendors

Easier customer adoption, migration and upgrades enabling more agility and revenue opportunities

Where Does DCML Go From Here

- Working group formed, first meeting in November
- Target draft specification for public comment release end of 2003
- First DCML implementations based on draft specification available early 2004
- Submission to standards body in 2004

dcml

Data Center Markup Language

Setting the Standard for Utility Computing