ORACLE®

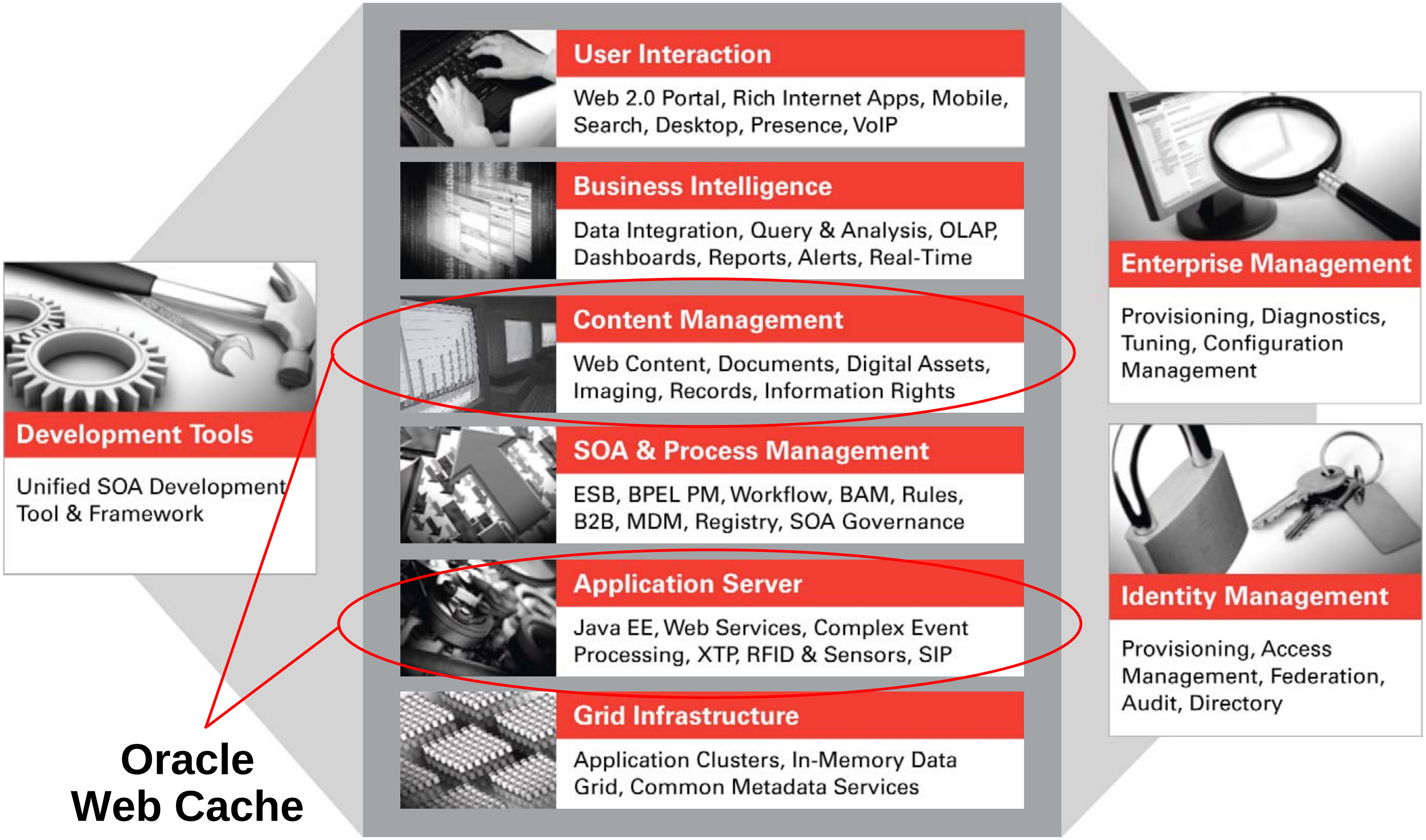
Oracle Web Cache 11g Overview

Oracle Web Cache

- Oracle Web Cache is a secure reverse proxy cache and a compression engine deployed between
  - Browser and HTTP server
  - Browser and Content Management server

  to improve the performance of web sites by caching frequently accessed content
- Oracle Web Cache supports
  - Static Content Caching
  - Dynamic Content Caching
  - Partial Page Caching
  - Request Filtering

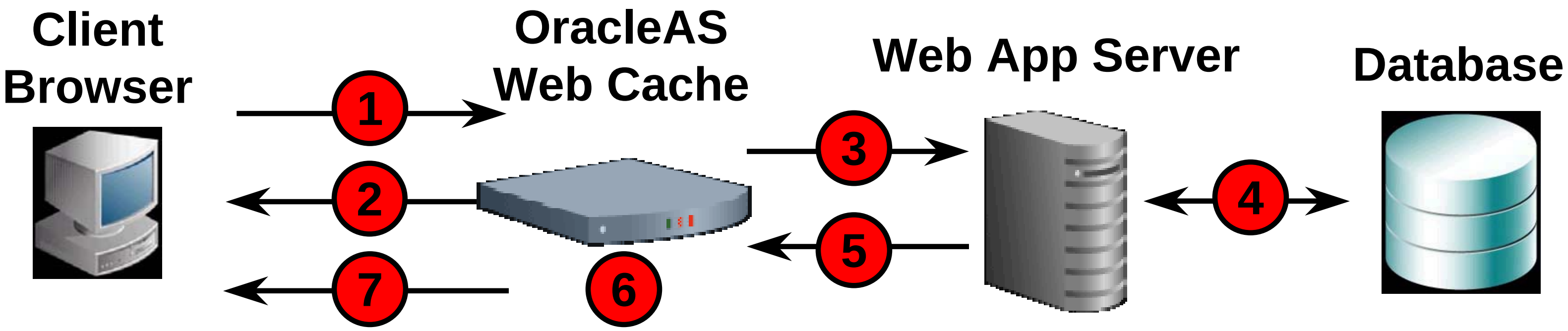
Oracle Fusion Middleware



User Interaction
Web 2.0 Portal, Rich Internet Apps, Mobile, Search, Desktop, Presence, VoIP
Business Intelligence
Data Integration, Query & Analysis, OLAP, Dashboards, Reports, Alerts, Real-Time
Content Management
Web Content, Documents, Digital Assets, Imaging, Records, Information Rights
SOA & Process Management
ESB, BPEL PM, Workflow, BAM, Rules, B2B, MDM, Registry, SOA Governance
Application Server
Java EE, Web Services, Complex Event Processing, XTP, RFID & Sensors, SIP
Grid Infrastructure
Application Clusters, In-Memory Data Grid, Common Metadata Services
Development Tools
Unified SOA Development Tool & Framework
Enterprise Management
Provisioning, Diagnostics, Tuning, Configuration Management
Identity Management
Provisioning, Access Management, Federation, Audit, Directory

Oracle Web Cache

How Oracle Web Cache Works



1. Client sends HTTP request
2. Web Cache responds immediately if cached object is available
3. If object is not in cache, Web Cache requests object from Application Server
4. Application Server generates response (may include Database queries)
5. Application Server responds to Web Cache
6. If response is cacheable, Web Cache retains a copy for subsequent requests
7. Web Cache compresses page and responds to Client

Oracle Web Cache

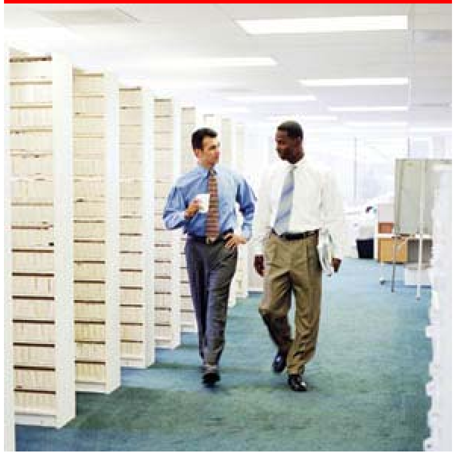
Key Features

- Significant Performance Improvement for your web applications
  - Accelerates web applications by serving cached documents from memory
  - Reduces load on content generating origin servers
  - Load Balancing requests across multiple origin servers
  - On the fly compression for cacheable and non-cacheable content
- Security
  - Request filtering to prevent malicious requests
    - Filter types: IP, URL, HTTP Method, Headers, Query String, and URL Format
  - SSO integration
    - Restrict access and caching to authenticated users only
  - SSL between browser and Web Cache, and SSL between Web Cache and origin servers for extra security

Oracle Web Cache

Key Features

- Ease of deploying and adding Oracle Web Cache to site topology
- Automated caching and invalidation based on response headers
- Comprehensive popular requests report
- Ease of configuring custom caching rules and invalidation rules
- Site level switch for caching and compression
- Common SSL configuration framework
- User specific statistics and performance monitoring
- Audit, event-based, and request-based logging

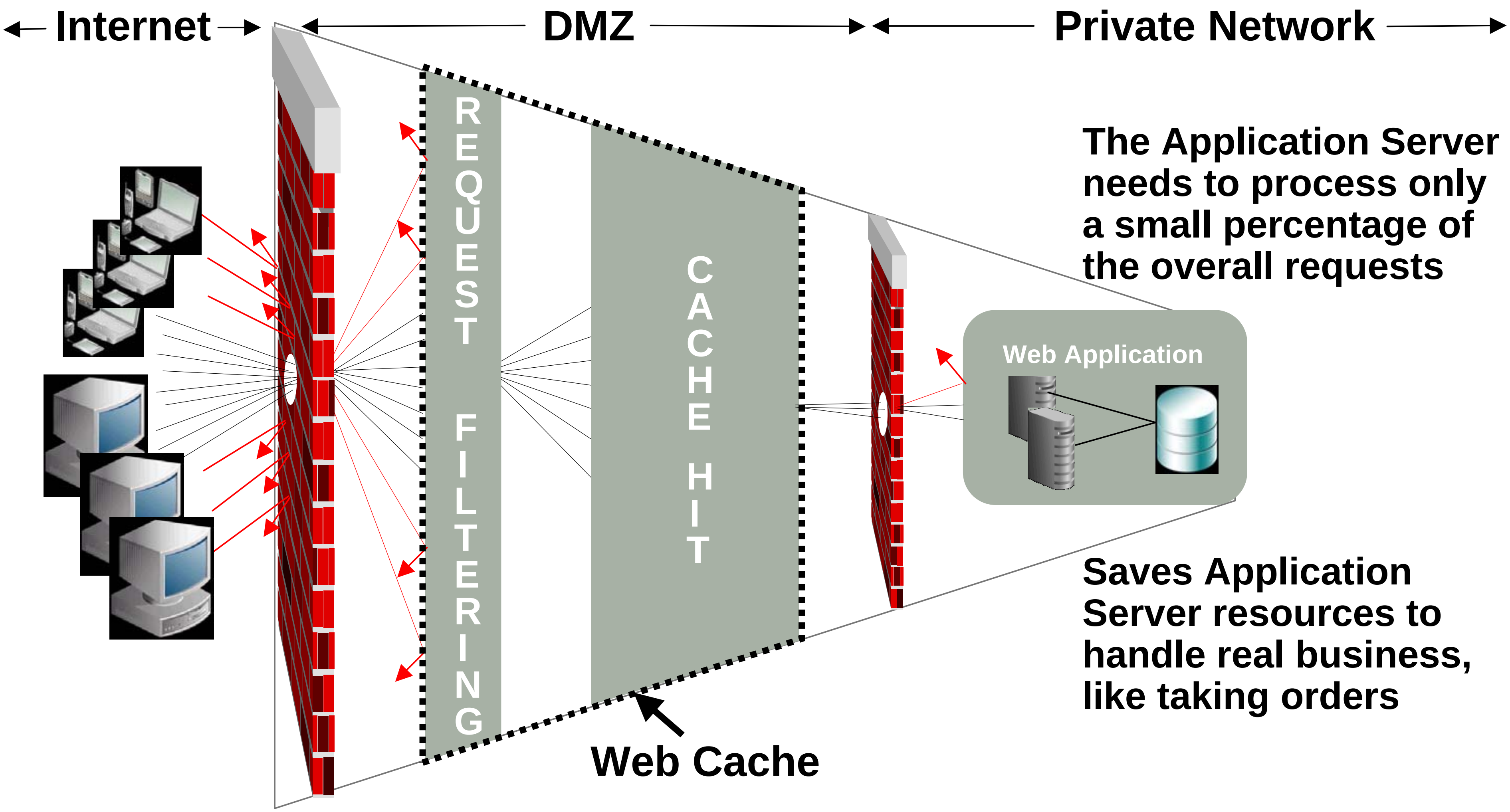


Request Filtering with Oracle Web Cache

Request Filtering with Oracle Web Cache

- Takes advantage of Web Cache’s position at the front of the HTTP application stack
- Stops illegal or malformed requests at the outer level before they reach the application server (based on user defined rules)
- Enforces legal requests such as correct URLs, correct session cookie values, etc (based on user defined rules)
- Has multiple filtering options
- Displays statistics alongside each filter rule to measure its actual effectiveness

Protecting Your Application Server with Oracle Web Cache



The Application Server needs to process only a small percentage of the overall requests

Saves Application Server resources to handle real business, like taking orders

Request Filtering

- Rejects illegal requests (black list)
- Enforces legal requests (white list)
- Suggests new rules by profiling actual application traffic
- Verifies new rules against traffic before activating
- Supports dynamic modification of filter rules without restarting
- Monitors filtering effectiveness in real time
- Allows customizing of response behavior for denied requests
- Allows customizing of audit settings for all requests

Black and White Listing

- Black listing is useful for blocking known bad requests (e.g. TRACE method). To use black listing:
  - Describe the illegal requests which should be denied
  - Set the “Catch All” rule to allow all other requests
- White listing provides more thorough filtering, but requires more knowledge of the application.To use white listing:
  - Enumerate all legal requests which should be allowed (Learned rules and monitor mode can help)
  - Set the “Catch All” rule to deny all other requests
- Black listing, white listing, or a hybrid approach are supported

Request Filter Types

- Client IP address
- HTTP Method (e.g. disallow TRACE)
- Cookies and other HTTP headers
- URL (path prefix, file extension or regular expression)
- Query String and POST body
- Format validation for the request URL and query string (e.g. proper encoding)
- Privileged IP address – bypasses all filters (e.g. administrator’s use)

Request Filtering Rule Sets

- Each application can have its own set of rules for each filter type
- Another set of rules can be defined for all other applications
- Rule sets can be copied from one application to another
- Rule set modifications are dynamic (no restart required)
- Web Cache proposes rules that it learns from actual traffic
- Rule sets are evaluated in order, until first match

Request Filtering Rules

Each Rule:

- Can specify deny or allow action
- Can specify what type of matching operation is to be used (prefix, substring, or regular expression)
- Can specify a URL expression as secondary match criteria. For example:
  - Must match method of “GET” and URL of prefix “/mystore”
  - Must match HTTP Header name “Cookie” and URL of regular expression “.*catalog[0-9]”
- Can be set to monitor-only mode (does not deny)
- Can be disabled without being deleted

Response Options for Denied Requests

- Request is denied when it matches a non-monitoring rule with “Deny” action
- Can specify the type of response:
  - 200 success status with apology page
  - 403 forbidden status
  - 404 file not found status
  - 500 internal server error status with apology page
  - Close connection

Audit Options for All Requests

- Requests can optionally be logged to the audit log for:
  - Requests that were denied
  - Requests that were allowed
  - Both denied and allowed requests

Conclusion

- Just when you thought you knew everything about Oracle Web Cache, we took a successful component and made it even better!
- Still has
  - The rich in-memory caching capabilities to offload many requests from the application servers
  - Invalidation interface to better control contents of the cache
  - Clustering and load balancing capabilities
- Now, request filtering adds
  - A wide variety of filtering options to further offload the application servers
  - A layer of security in the web tier
  - Runtime statistics alongside the configuration
- Has many other new features and enhancements

ORACLE®