

**ORACLE®**



**ORACLE®**

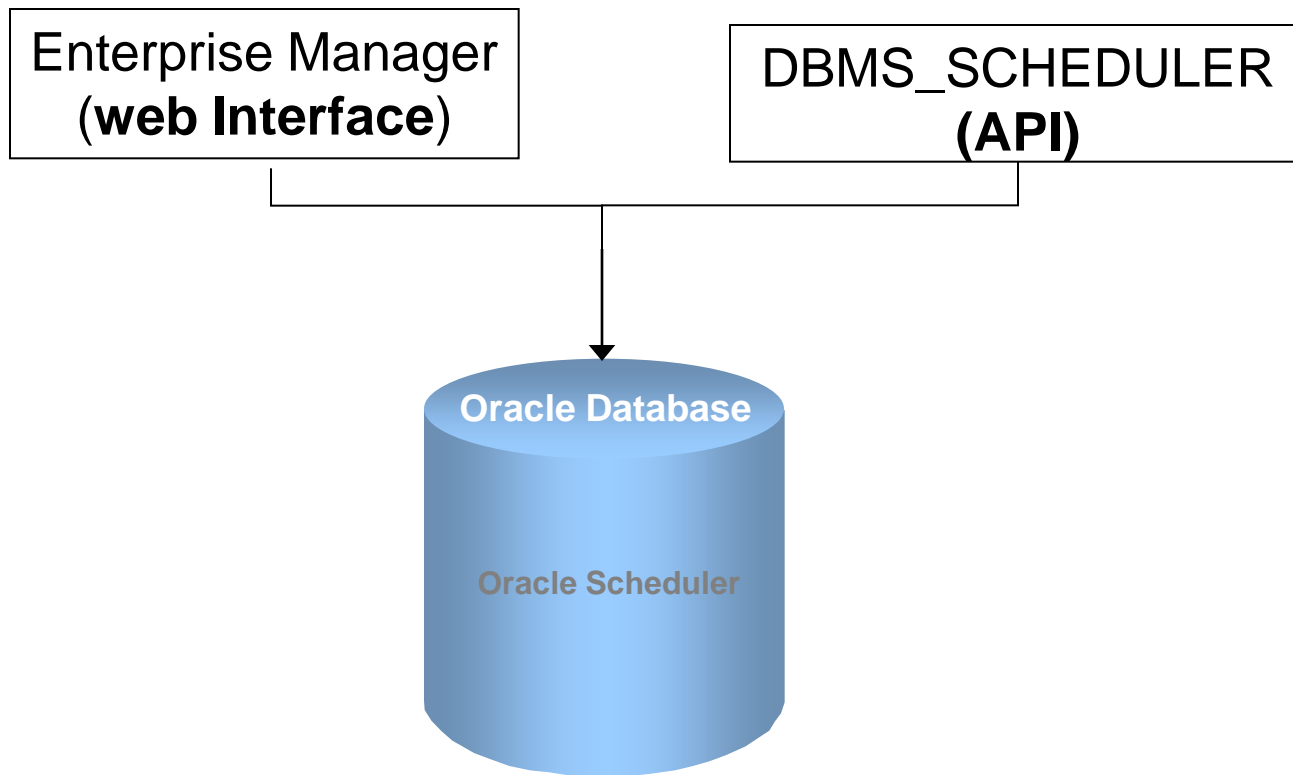
**Get the best out of Oracle Scheduler: Learn how you can leverage Scheduler for enterprise scheduling**

Vira Goorah (vira.goorah@oracle.com) – Oracle

# Agenda

- What is Oracle Scheduler?
- Why use Oracle Scheduler?
- What can it do?
- Architecture
- Managing and Monitoring
- Best Practices
- Customer Case Study
- Q&A

# Oracle Scheduler Overview



# Why Oracle Scheduler?

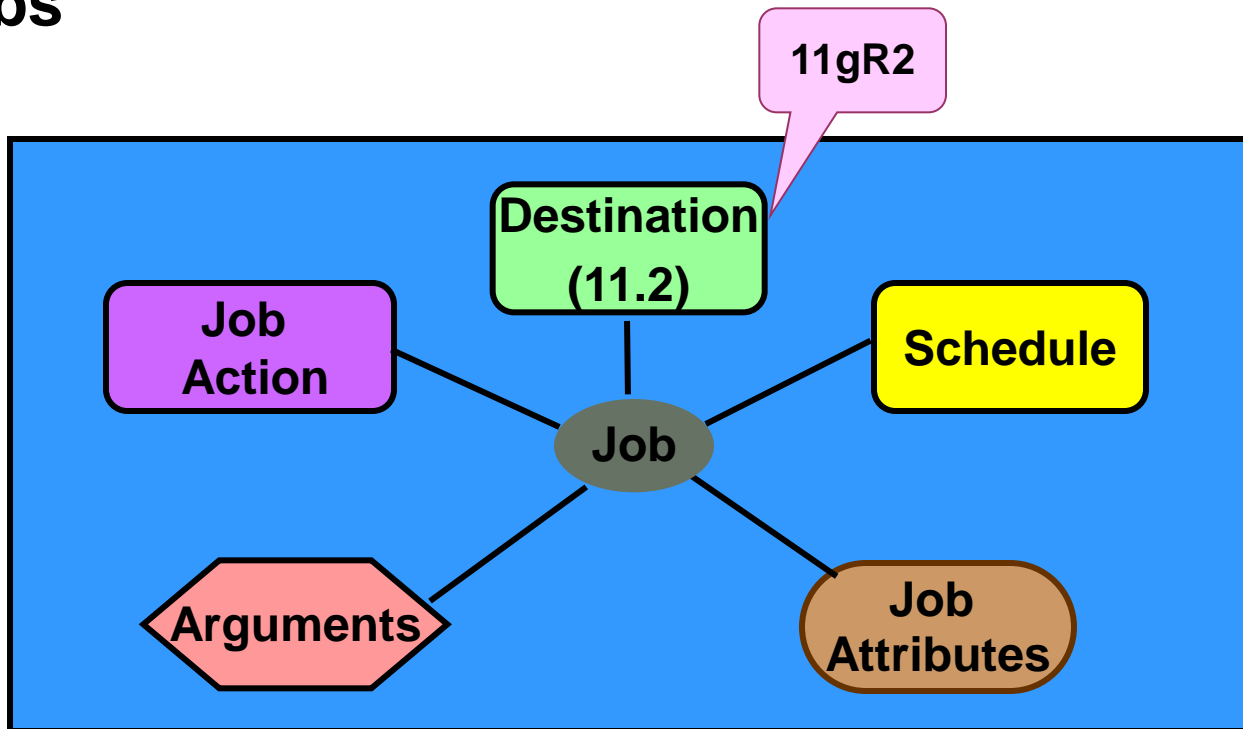
## Top 10 Reasons

- 1 Save money**
- 2 Leverage existing DB knowledge
- 3 DBA can control jobs
- 4 Manage jobs similarly on all platforms
- 5 Easily move jobs between systems
- 6 Jobs run only when the database is up
- 7 Security
- 8 High availability and Recovery
- 9 Scalability
- 10 Exploit new DB features

# What can it do?

- Job types supported
  - PL/SQL blocks & procedures, executables (OS jobs)
- Execute jobs based on:
  - Time – Time-based scheduling
  - Outcome of other jobs – Dependency scheduling
  - Event – Event-based scheduling
- Prioritize jobs
  - Align job processing with business req.

# Jobs



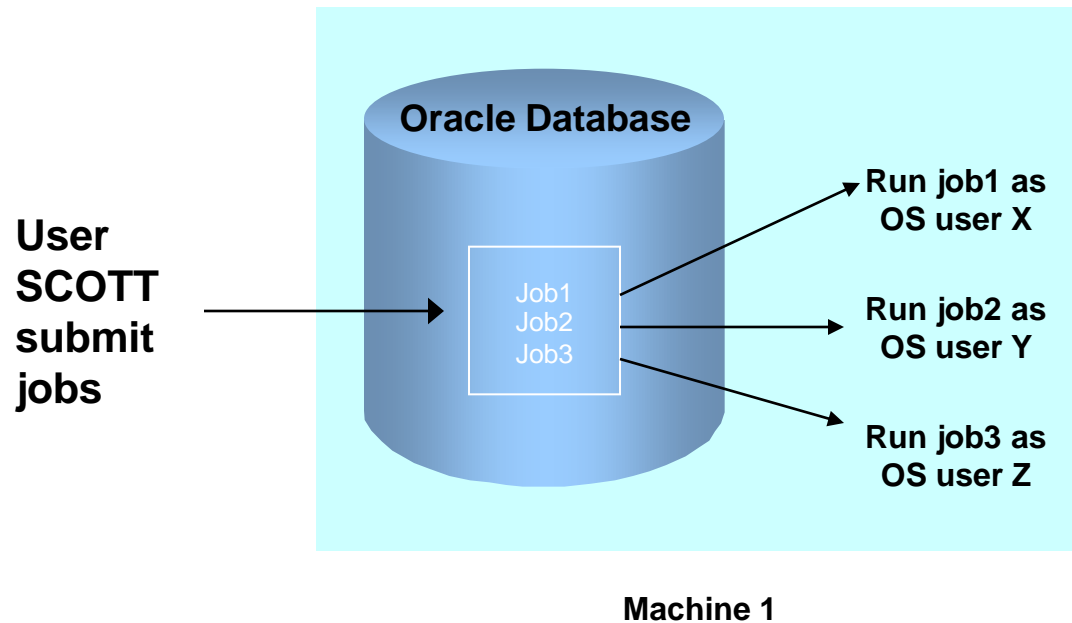
- PL/SQL blocks
- PL/SQL/Java procedures
- Executables

# Supported Jobs

- PL/SQL blocks, PL/SQL Procedures, Executables
- Executables (OS jobs) are jobs running outside the database. For example shell scripts.
  - As of v11gR1
    - OS jobs as different OS users
    - OS jobs on remote machines

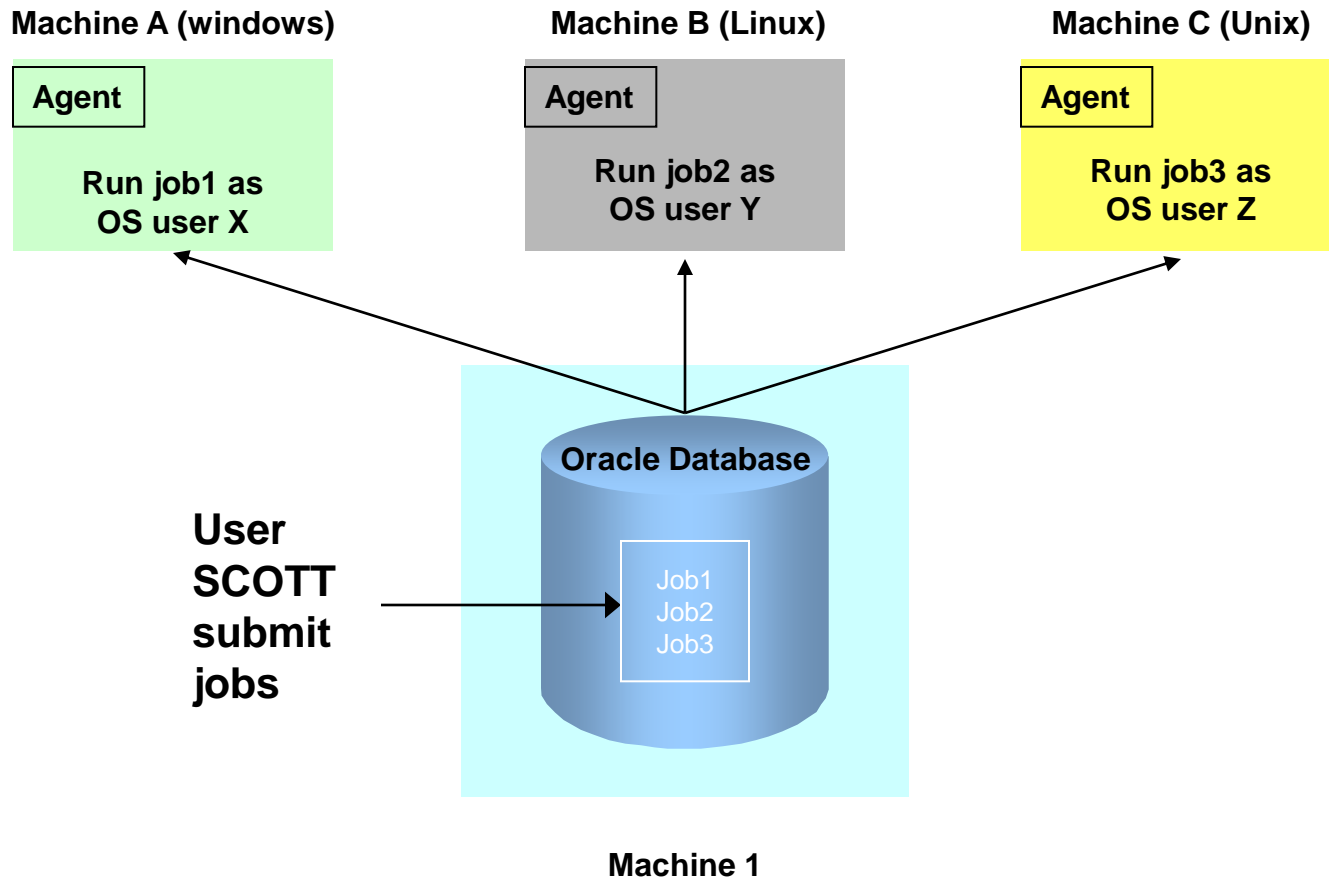


# Local OS jobs as different OS users



# OS jobs on remote machines

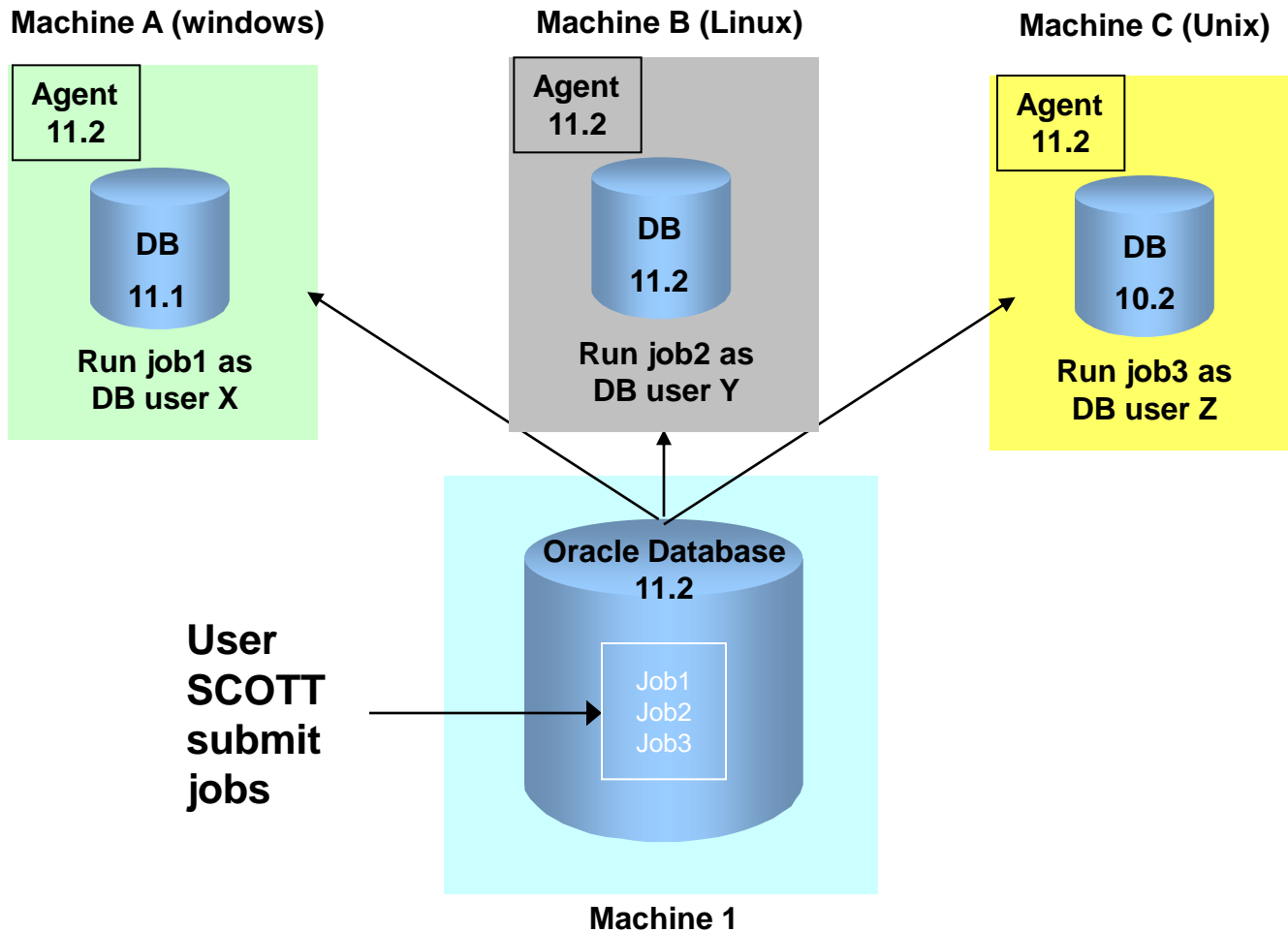
## Remote External Jobs



# Database jobs on remote machines

## Remote Database Jobs

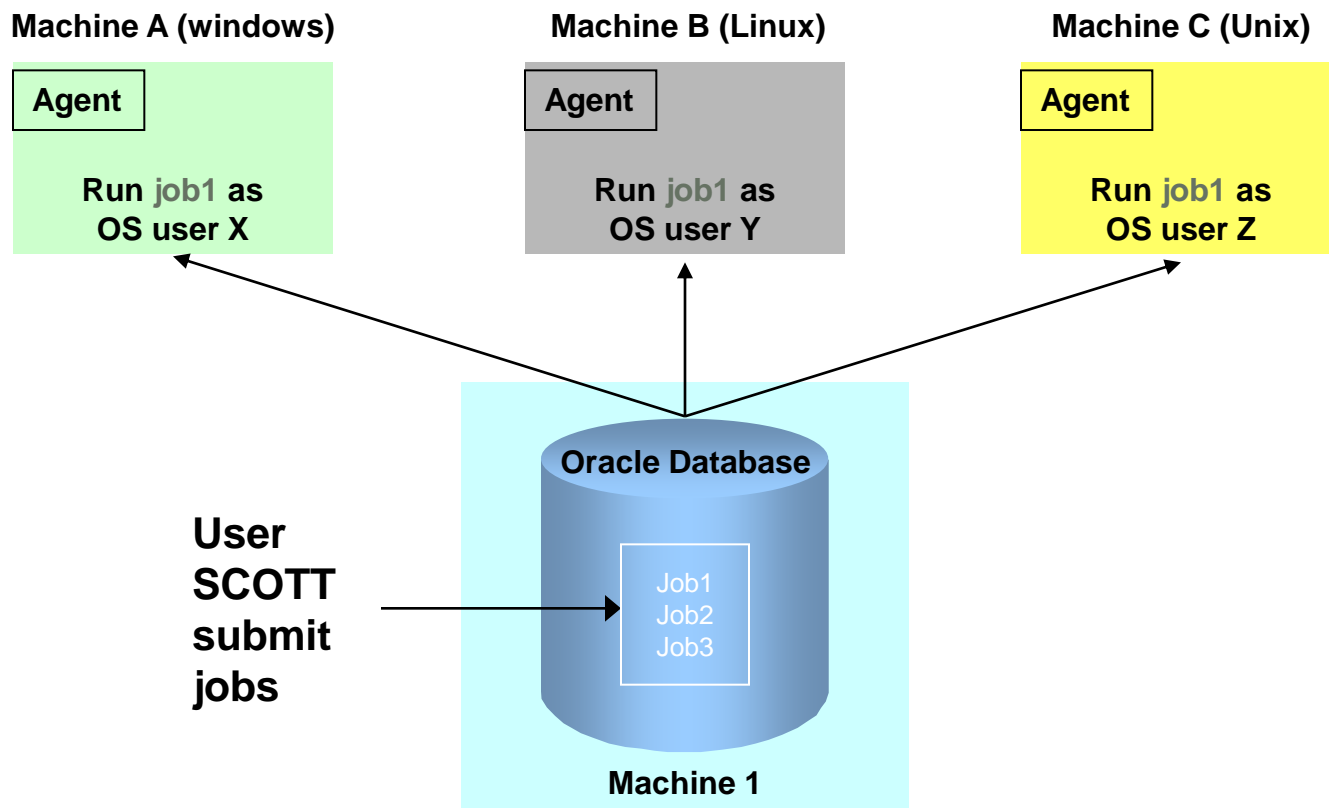
11gR2



# Multi-Destination Jobs

11gR2

Same job running on different destinations with different credentials



# Multi-Destination Jobs

11gR2

- Each individual destination job is a child job of the parent multi-destination job child created by the parent at run time
- The parent multi-destination job fails if all of its child jobs failed
- If one or more, but not all child jobs fail then the parent job will get the status 'partially failed'
- In repeating multi-destination jobs the parent job will skip those child destinations that are still working on their previous run

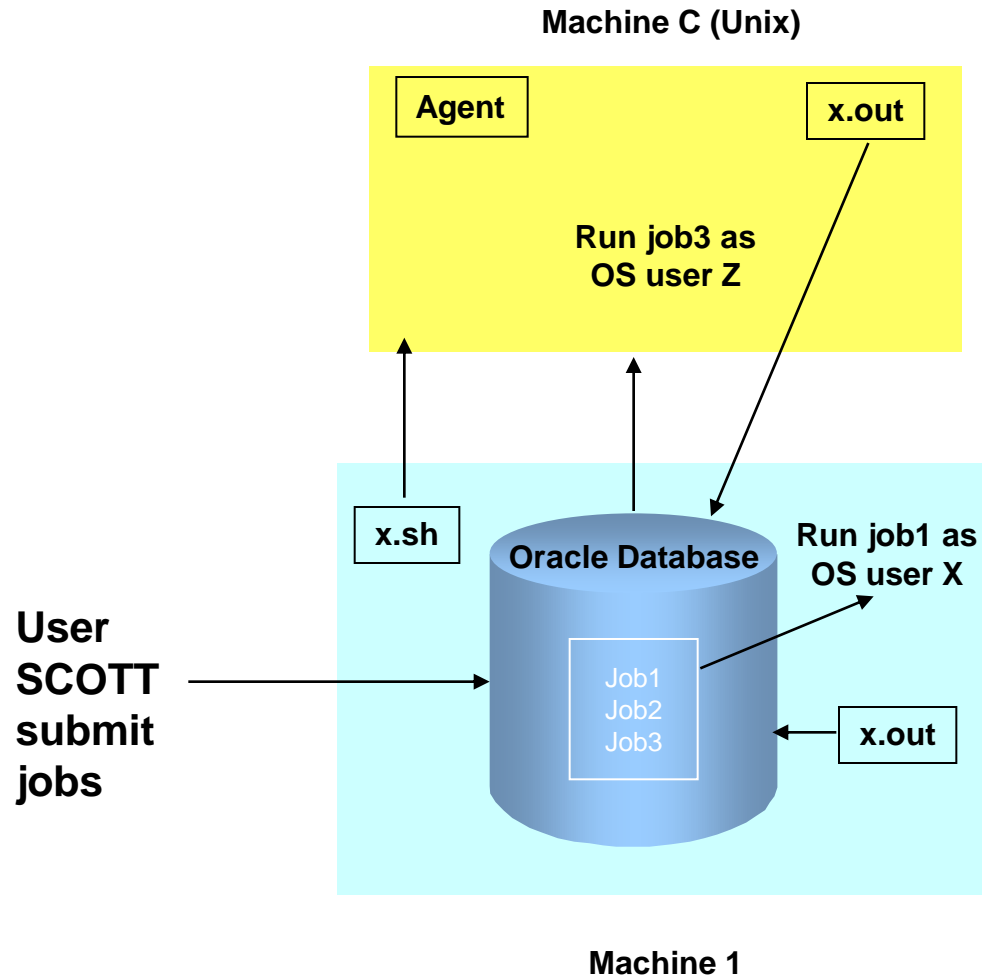
- Destinations are database objects
- Displayed in `*_SCHEDULER_DESTS` view
- 2 types of destinations
  - DB destinations – Use `CREATE_DATABASE_DESTINATION`
    - Attributes: name, agent\_name and tns\_name
    - Displayed in `*_SCHEDULER_DB_DESTS` view
  - External destinations - Created automatically when agent is registered
    - Destination object will have the same name as the agent
    - Attributes: name, hostname/IP address, Port
    - Displayed in `*_SCHEDULER_EXTERNAL_DESTS` view

# Destination Groups

11gR2

- Database object which is a list of named destinations
- Use destination groups to execute jobs at multiple destinations
- 2 types of destination groups –All members must be of the same type
  - Database destination groups
  - External destination groups
- Format of destination members is  
`[[schema.]credential@][schema.]destination  
member => 'LOCAL',CREDX@HOST1,HOST2'`
  - Credential is optional. If not specified will use the `credential_name` associated with the job

# File Transfer





# Scheduler Agent

- Agent is multi-threaded and java-based
- Secure communication between the database and the agent
  - Communication to agent is HTTPS
  - Results are sent back using HTTP (XMLDB HTTP server)
  - Job requests to agent and file transfers to and from agent are encrypted. As of 11gR2, job results are also encrypted
  - Uses certificates for security
  - Connection open only to send execution request (not duration of job)
- Advanced users can customize configuration File
  - Agent name, Max running jobs, authorized users and logging level (default DB – 11.2)

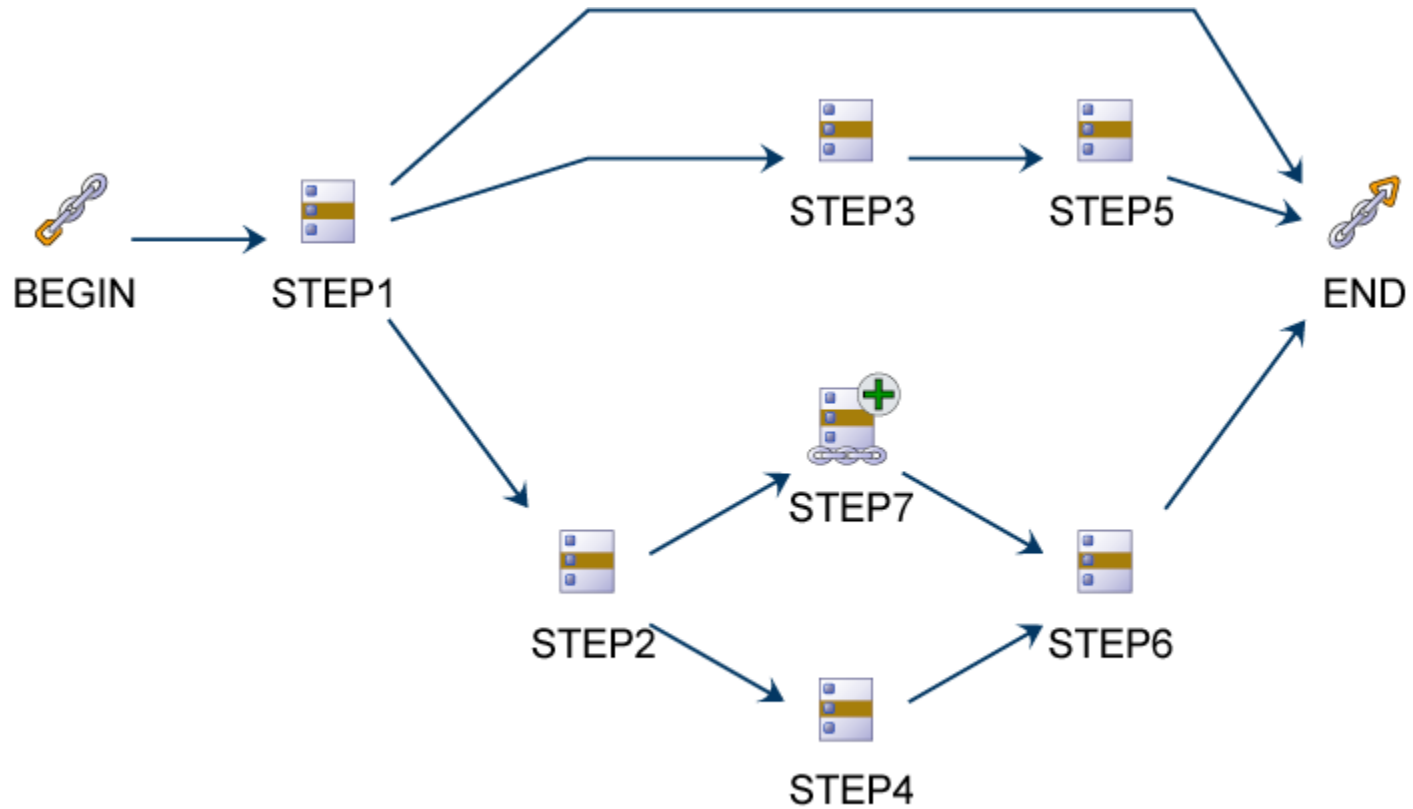
# Scheduling Flexibility

- Time-based
  - Basic day/time (timezone support)
  - Custom calendar
- Outcome of other jobs
  - Job dependency scheduling
- Any event
  - Server generated alerts
  - Events external to the database – file arrival

# Time-Based Scheduling

- Specify advanced repeat schedules
  - Last sun of month, every day at 4,5 and 6 PM
- Composite Schedules
  - Use existing schedules to create new schedules (Include, Exclude, Intersect)
- Business Calendars
  - Define custom calendars e.g. Fiscal calendar and Holiday calendar

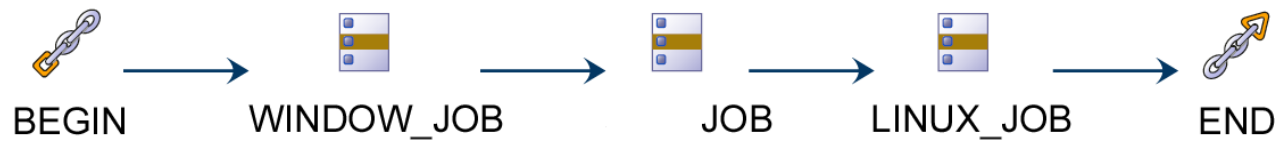
# Dependency Scheduling




# Dependency Scheduling

- Execution of a step depends on other step(s)
- Chain can include: Event steps, another chain, time restrictions
- Conditions: State, Error code, Start/End date etc. Can include AND, OR, NOT
- Action examples: Start, Start with delay, Stop
- Multiple end conditions
- Pause chain, run chain, skip steps
- Scheduling of a chain

# Multiplatform Chain



# Event-based Scheduling

- Execute jobs based on any event
  - Server generated alerts
  - External events such as a file arrives on the system
    - File Watcher  11gR2
- Events can be part of a chain

- Used to trigger a job when a file arrives on the system (local or remote)
- Can specify the following attributes:
  - Directory path - Can be relative to the ORACLE\_HOME (or agent\_home in the case of remote)
  - File Name - Can have wildcards anywhere in the name
  - Destination name for remote systems (null for local)
  - Min file size
  - Steady state duration
- If the destination is local a system job runs periodically to check for files specified by the different file watcher objects (For remote – the agent checks)



# Monitoring and Managing

- Take action on job state changes
  - For example get notification when a job fails
  - Job started, completed, succeeded, failed, stopped, disabled, chain stalled ....
- Filter by job attributes - owner, status, name
- Manage jobs at group level
- Logging: Full - Audit trail, Runs only, **Errors only**, Off
- Set logging level - per job class and per job (can only increase the level of logging)
- Purging of logs

# Email Notification

11gR2

- Receive notifications for specified job state events
  - For example - Job failed, Job completed, Job stopped etc.
- Emails can be sent to a list of email addresses
  - Emails can be sent to different addresses for different states
- Can specify a sender for the emails
- Subject and body default templates
- Need to specify the outgoing SMTP server

# Best Practices – Top 10 recommendations

1. Use caution when disabling Oracle Scheduler
2. Don't create Scheduler objects in SYS schema
  - Don't modify objects in SYS schema unless advised
3. Use region name for default timezone (follow DST)
4. Use credentials for OS jobs as of 11gR1
  - Use Transparent Data Encryption for the password col.
5. Use caution with `CREATE EXTERNAL JOB` and `CREATE ANY JOB` privileges
  - Upgrade script from 10gR1 automatically grants `CREATE EXTERNAL JOB` to users with `CREATE JOB`
  - `CREATE ANY JOB` – can run jobs as any DB user

## Best Practices – Top 10 recommendations

6. Put frequently running jobs in a job class with logging set to “failed runs only”
7. Use services instead of instance ID in RAC environment
  - Jobs are load balanced across and maintenance is easier
  - Use instance ID ONLY if job needs to run on that instance
8. Agent
  - Set agent name in the agent configuration file
  - Disable file transfers/ job execution if you don't need it
9. Only use Windows as the job schedule when a particular resources is needed to run the job
10. Set default sender for email notification – 11gR2

# Q & A

**ORACLE®**