The DOT Service Shall…

- Utilize the DOT library
- Support iGrid-10 model-based paradigm
- Minimize a scalar-valued function
- Maximize a scalar-valued function
- Enforce scalar-valued inequality constraints
- Enforce scalar-valued equality constraints
- Perform unconstrained optimization
- Accept user configuration via ASCII file
- Accept user configuration via API
- Default to standard configuration in absence of user configuration

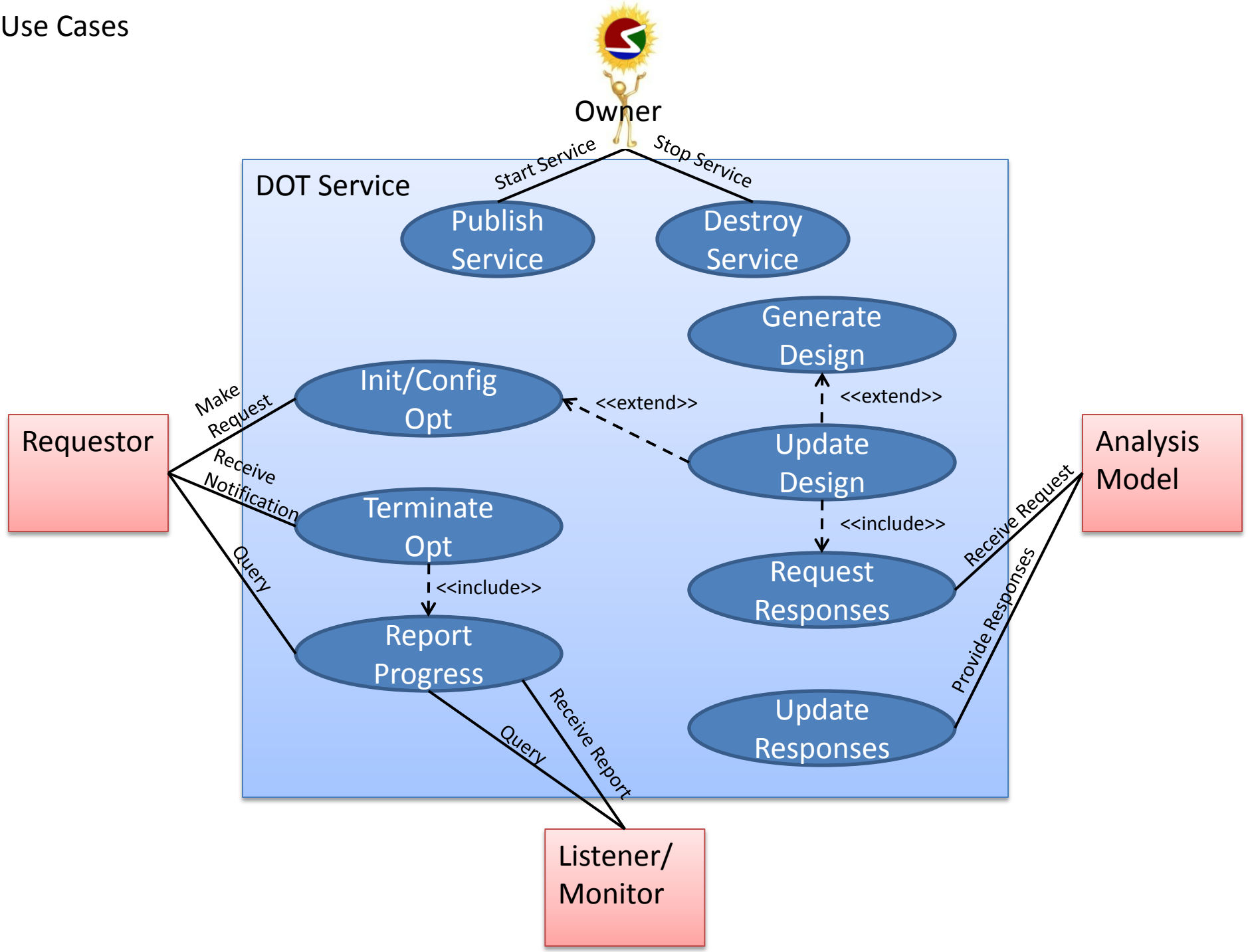
The DOT Service Shall… (cont)

- Return the identities of the objective and constraints to be evaluated
- Return the identities of gradients to be evaluated
- Return values of design variables at which to evaluate
- Accept values of objective and constraints requested
- Accept values of gradients requested
- Access run state via API
- Provide feedback mechanism to allow progress reporting (interface callback/Job Monitor tie-in/file URL)
- Report to a comma-delimited file
- Report to a Tecplot formatted file
- Report to a gnuplot formatted file
- Allow user selection of no, one, or multiple file formats

The DOT Service Shall… (cont)

- Return DOT library output files in entirety
- Organize output files in uniquely identifiable directory
- Report the optimized design back to the requestor
- Perform multiple optimizations in series without requiring service republishing
- Optionally provide diagnostic information
- Halt the optimization process if notification is received from an analysis provider that a design is inadmissible

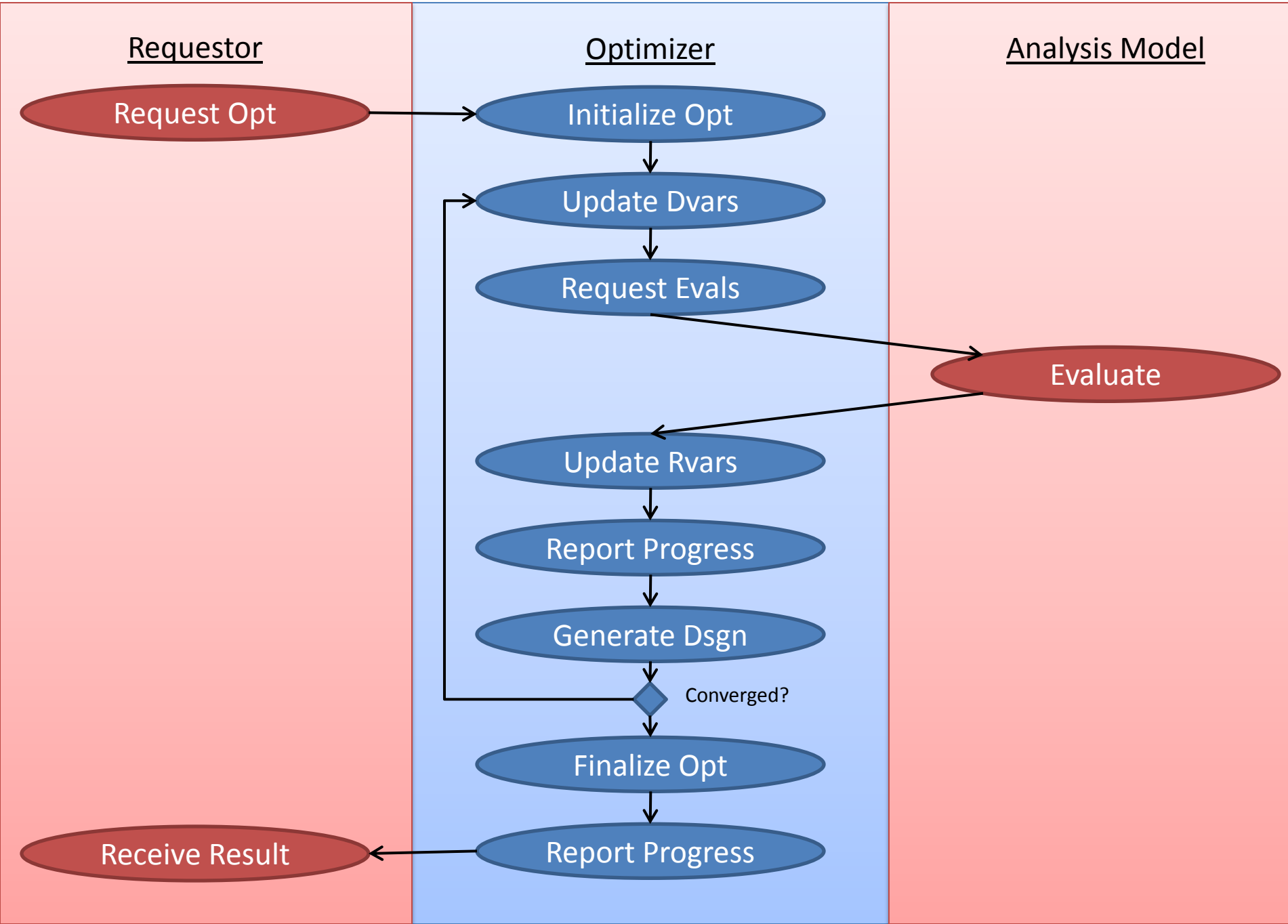
Use Cases


Owner
Start Service
Stop Service
DOT Service
Publish Service
Destroy Service
Generate Design
Init/Config Opt
<<extend>>
<<extend>>
Update Design
Requestor
Make Request
Receive Notification
Terminate Opt
<<include>>
<<include>>
Request Responses
Analysis Model
Receive Request
Query
Report Progress
Provide Responses
Update Responses
Query
Receive Report
Listener/ Monitor

Data Flow

External	Opt Dir, DOT Props, Obj, Cons, Dvars			Rvar Values, Grad Values			
	Initiate/ Configure Optimization	Initial Design	Evaluators		Configs, Constraints, Dvars, Limits, DOT Library		Cnvg Criteria, Iter Limits
		Update Design	Dvars			Dvar Values	
			Request Responses	Rvars			
				Update Responses	Responses, Gradients	Rvar Values, Grad Values	
		Dvar Values	Responses, Gradients Needed		Generate New Design		
						Report Progress	Reply to Request
							Terminate Optimization
External			Evaluation Requests			Intermediate Data	Data Files, Final Design, Opt Result

Activities


Requestor
Optimizer
Analysis Model
Request Opt
Initialize Opt
Update Dvars
Request Evals
Evaluate
Update Rvars
Report Progress
Generate Dsgn
Converged?
Finalize Opt
Report Progress
Receive Result

Init/Config Optimization

- Create DOT Data Directory
- Open Output Files
- Instantiate Opti Object
  - Load DOT Library
- Parse DOT Config Params
- Identify Objective
- Identify Constraints
- Identify Design Vars

Update Design Variables

- Synchronize Model Design Variables and DOT Design Variable Vector

Request Evaluations

- Identify and Mark Responses Requiring Evaluation
- Signal Evaluation Request

Update Response Variables

- Synchronize Model Response Variables and DOT Objective and Constraint Vectors

Report Progress

- Send optimizer state, design variable, objective, constraint, and/or gradient information to logger, console, file, monitor, listener, and/or return context

Generate Design

- Call DOT Library
- Check Convergence

Finalize Optimization

- Report Progress
- Set Opti Object to null
  - Removes Reference to DOT Library
- Close Output Files
- Copy Output to DOT Data Directory
- Set DOT Data Directory to null
  - No Further Access to Directory Contents

Class Diagram