Glossary

abstract syntax ..... A conceptual data organization. cf. *concrete syntax.*

action methods ..... The methods initialize(), prefire(), fire(), postfire(), and wrapup() in the Executable interface.

actor ..... An executable entity. This was called a *block* in Ptolemy Classic.

anytype ..... The Ptolemy Classic name for *undeclared type*.

applet ..... A Java program that is downloaded from a web server by a browser and executed in the client's computer (usually within a plug-in for the browser). An applet has restricted access to local resources for security reasons. cf. application.

application ..... A Java program that is executed as an ordinary program on a host computer. Unlike an applet, an application can have full access to local resources such as the file system. cf. applet.

atomic actor ..... A primitive actor. That is, one that is not a composite actor. This was called a *star* in Ptolemy Classic.

attribute ..... A named property associated with a named object in Ptolemy II. Also, in XML, a modifier to an element.

block ..... The Ptolemy Classic name for an *actor.*

browser ..... A program that renders HTML and accesses the worldwide web using the HTTP protocol.

channel ..... A path from an output port to an input port (via relations) that can transport a single stream of tokens.

clustered graph ..... A graph with hierarchy. Ptolemy II topologies are clustered graphs.

code generation ..... Translation of a model into efficient, standalone software for execution autonomously from the design environment. Code generation was a major emphasis of Ptolemy Classic.

composite actor ..... An actor that is internally composed of other actors and relations. This was called a *galaxy* in Ptolemy Classic.

concrete syntax ..... A persistent representation of a data organization. cf. *abstract syntax.*

connection ..... A path from one port to another via relations and possibly transparent ports. A connection consists of one or more *relations* and two or more *links*.

container ..... An object that logically owns another. A Ptolemy II object can have at most one container.

dangling relation ..... A relation with only input ports or only output ports linked to it.

data polymorphism ..... Ability to operate with more than one token type.

deep traversals ..... Traversals of a clustered graph that see through transparent cluster boundaries (transparent composite entities and ports).

disconnected port ..... A port with no relation linked to it.

director ..... An object that controls the execution of a model or an opaque composite entity according to some *model of computation*.

domain ..... An implementation of a model of computation in Ptolemy II and Ptolemy Classic.

domain polymorphism ..... Ability to operate under more than one model of computation.

element ..... In XML, a portion of a document consisting of a begin tag, a body, and an end tag.

entity ..... A node in a Ptolemy II clustered graph. Also, in XML, a named text segment.

event ..... In the DE domain, an event is a token with a time stamp.

execution ..... One invocation of initialize(), followed by any number of *iterations*, followed by one invocation of wrapup().

executive director ..... From the perspective of an actor inside an opaque composite actor, the director of the container of the opaque composite actor.

galaxy ..... The Ptolemy Classic name for a *composite actor*.

immutable property ..... A property of an object that is set up when the object is constructed and that cannot be changed during the lifetime of the object.

iteration ..... One invocation of prefire(), followed by any number of invocations of fire(), followed by one invocation of postfire().

link ..... An association between a port and a relation.

manager ..... The top-level controller for the execution of a model.

model ..... A complete Ptolemy II application. This was called a *universe* in Ptolemy Classic.

model of computation ..... The rules that govern the interaction, communication, and control flow of a set of components.

MoML ..... Modeling markup language, an XML dialect for specifying component-based designs such those in Ptolemy II.

multiport ..... A port that can send or receive tokens over more than one channel.

opaque ..... For a composite entity or a port, an attribute that indicates that the inside should not be visible from the outside. That is, deep traversals of the topology do not see through an opaque boundary.

opaque composite actor ... A composite actor with a local director. Such an actor appears to the outside domain to be atomic, but internally is composed of an interconnection of other actors. This was called a *wormhole* in Ptolemy Classic.

package ..... A collection of classes that forms a logical unit and occupies one directory in the source code tree.

parameter ..... An *attribute* with a value. This was called a *state* in Ptolemy Classic.

particle ..... The Ptolemy Classic name for a *token*.

port ..... A named interface of an entity to which connections be made.

Ptolemy Classic ..... A C++ software system for construction of concurrent models and implementation through code generation.

Ptolemy II ..... A Java software system for construction and execution of concurrent models.

Ptolemy Project ..... A research project at Berkeley that investigates modeling, simulation, and design of concurrent, networked, embedded systems.

relation ..... An object representing an interconnection between entities.

resolved type ..... A type for a port that is consistent with the type constraints of the actor and any port it is connected to. It is the result of type resolution.

servlet ..... A Java program that is executed on a web server and that produces results viewed remotely on a web browser.

star ..... The Ptolemy Classic name for an *atomic actor*.

state ..... The Ptolemy Classic name for a *parameter*.

subpackage ..... A package that is logically related to a parent package and occupies a subdirectory within the parent package in the source code tree.

tag ..... In XML, a portion of markup having the syntax <*tagname*>.

token ..... A unit of data that is communicated by actors. This was called a *particle* in Ptolemy Classic.

topology ..... The structure of interconnections between entities (via relations) in a Ptolemy II model. See *clustered graph*.

transparent ..... For an entity or port, not opaque. That is, deep traversals of the topology pass right through its boundaries.

transparent composite actor
A composite actor with no local director.

transparent port ..... The port of a transparent composite entity. Deep traversals of the topology see right through such a port.

type constraints ..... The declared constraints on the token types that an actor can work with.

type resolution ..... The process of reconciling type constraints prior to running a model.

undeclared type ..... Capable of working with any type of token. This was called *anytype* in Ptolemy Classic.

universe ..... The Ptolemy Classic name for a *model*.

width of a port ..... The sum of the widths of the relations linked to it, or zero if there are none.

width of a relation ..... The number of channels supported by the relation.

wormhole ..... The Ptolemy Classic name for an *opaque composite actor*.