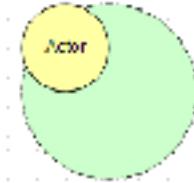


(Optional) i* Refresher

i* is a high-level socio-technical modeling language, meant to answer: who? why? why not? how? how else? and what? It consists of actors, various elements, and links.



An i* actor: a person, role or organization.

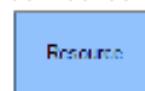
Elements



A (hard) goal: something clear-cut that an actor wants to achieve



A task: an action or step that an actor wants to or must perform

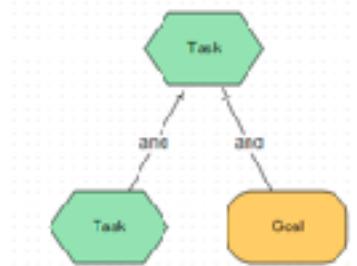


A soft goal: a goal an actor wants to achieve, but of a less clear-cut, fuzzy nature. Usually used to represent qualities or criteria for decisions

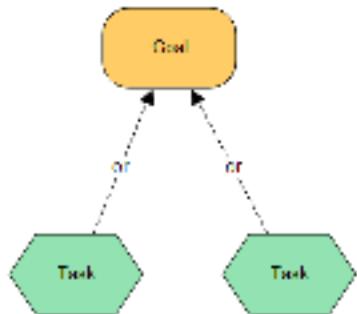


Links

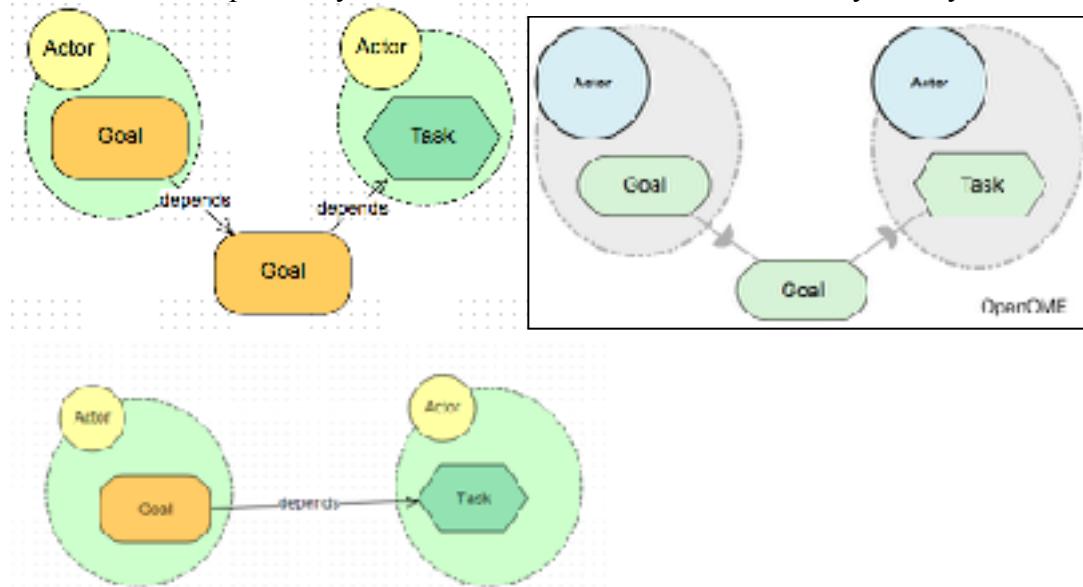
And Decomposition: to achieve the parent, all the children must be achieved



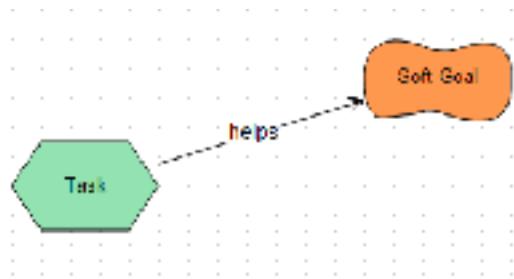
Or Decomposition (**Means-Ends**): to achieve the parent, one or more alternatives must be achieved



Dependency: An actor depends on something from another actor, usually to achieve some goal or task. Note: Dependency links look different than in other tools you may have used.



Contribution: an element in an actor has a positive or negative affect on some soft goals. The affect can be strongly positive (makes), weakly positive (helps), weakly negative (hurts), or strongly negative (breaks).



Study Terminology and Reference

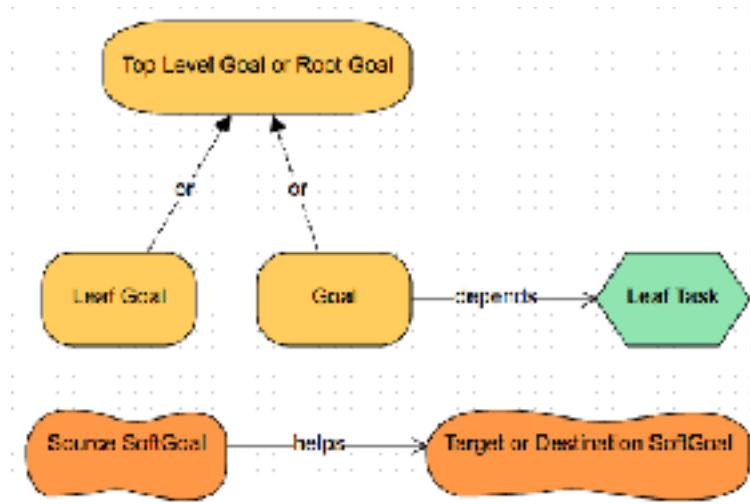
Here we introduce common terminology that is used to discuss i* models.

Leaf Nodes and Root (Top-Level) Nodes

i* models often form tree structures.

*** **Leaf Nodes** (Goals, Tasks, Resources, Soft Goals) are defined as nodes **with NO Incoming** decomposition (AND/OR [Means-End]) or contribution links (Makes, Breaks, Helps, Hurts) and **NO Outgoing** dependency links (Depends).

*** **Root (Top-Level) Nodes** (Goals, Tasks, Resources, Soft Goals) are defined as nodes **with NO Outgoing** decomposition (AND/OR [Means-End]) or contribution links (Makes, Breaks, Helps, Hurts) and **NO Incoming** dependency links (Depends).



Note: The *Source SoftGoal* would be a Leaf Node and the *Target or Destination SoftGoal* would be a Root (Top-Level) Node.

Evaluation Labels

To evaluate goal models we use the following evaluation labels.

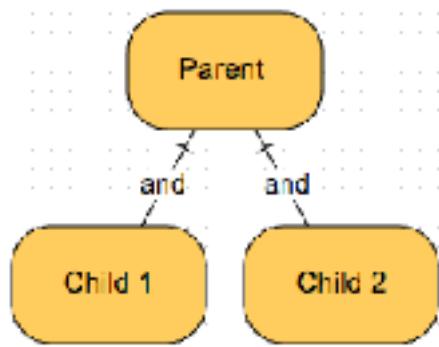
Denied ✗ ✓ Satisfied
Partially Denied ✗ ✗ Partially Satisfied

Note: Unknown and Conflict may result from analysis.

Conflict ≥ ? Unknown

Forward Analysis Propagation Rules

And Decomposition: to achieve the parent, all the children must be achieved



For AND-Decomposition we take the minimum value of all the child nodes.

$$\checkmark < \checkmark < \times < \times$$

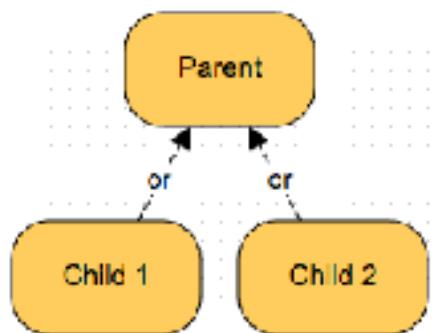
AND-Decomposition

Inputs		Result
Child 1	Child 2	Parent
✓	✓	✓
✓	✗	✗
✗	✗	✗
✓	✗	✗
✓	✓	✓
✓	✗	✓
✗	✗	✗
✓	✗	✗
✗	✓	✗
✗	✗	✗
✗	✓	✗
✗	✗	✗
✗	✓	✗
✗	✗	✗
✗	✗	✗
✗	✗	✗

OR-Decomposition

Inputs		Result
Child 1	Child 2	Parent
✓	✓	✓
✓	✗	✓
✗	✗	✗
✓	✗	✓
✗	✓	✓
✗	✗	✗
✓	✓	✓
✓	✗	✓
✗	✓	✓
✗	✗	✗
✓	✗	✓
✗	✓	✓
✗	✗	✗
✓	✓	✓
✓	✗	✓
✗	✓	✓
✗	✗	✗

Or Decomposition (Means-Ends): to achieve the parent, one or more alternatives must be achieved

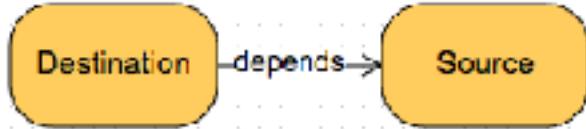


For OR-Decomposition we take the maximum value of all the child nodes.

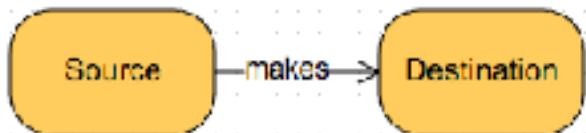
$$\checkmark < \checkmark < \times < \times$$

Forward Analysis Propagation Rules Continued

Dependency: Dependency links propagate the value in the opposite direction of the depends arrow. For example, the *Destination* depends on the *Source*. The label at *Source* is propagated directly to the *Destination*, see the table below.

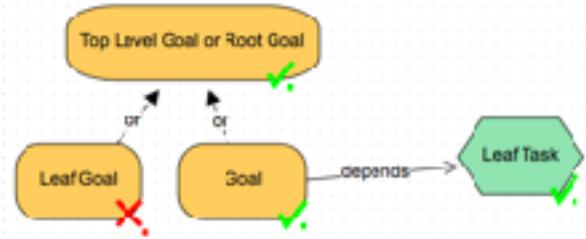


Contribution Links: Contribution links propagate based on the type of link as shown in this table below.



Source	Destination for each Link Type				
	Depends	Makes	Helps	Hurts	Breaks
✓	✓	✓	✓	✗	✗
✓, ✓	✓, ✓	✓, ✓	✓, ✓	✗, ✗	✗, ✗
✗, ✗	✗, ✗	✗, ✗	✗, ✗	✓, ✓	✓, ✓
✗	✗	✗	✗	✓	✓

Example:



For example, *Top Level Goal or Root Goal* gets the value of Partially Satisfied because in an OR-Decomposition we take the maximum of Partially Satisfied and Partially Denied.

Note: The value of *Goal* was first propagated from the value of *Leaf Task*.

Stochastic i* Models (Models that Change over Time)

We extend i* to enable analysis where the Leaf Nodes in the model change over time.

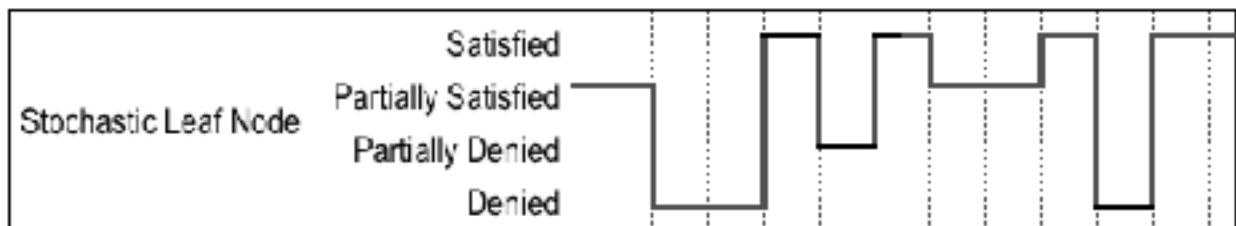
Recall to evaluate goal models we use the following evaluation labels.

Denied Satisfied
Partially Denied Partially Satisfied

Note: Unknown and Conflict labels are not used for Leaf Nodes in analysis.

If the Leaf Nodes change *Stochastically* over time, this means that in the next state they can have any of the evaluation labels including the same one as the previous state.

Below is an example of the how a Stochastic Leaf Node might change over time.



Dynamic i* Models (Models that Change over Time)

We extend i* to enable analysis where the Leaf Nodes in the model change over time.

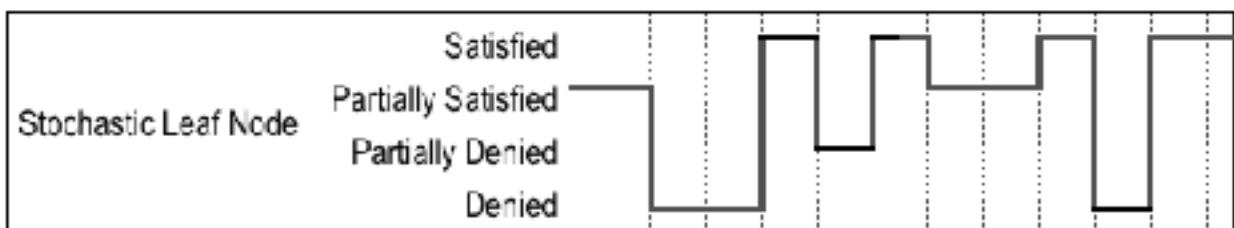
Recall to evaluate goal models we use the following evaluation labels.

Denied Satisfied
Partially Denied Partially Satisfied

Note: Unknown and Conflict labels are not used for Leaf Nodes in analysis.

If the Leaf Nodes change *Stochastically* over time, this means that in the next state they can have any of the evaluation labels including the same one as the previous state.

Below is an example of the how a Stochastic Leaf Node might change over time.



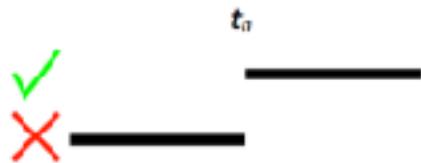
The next pages we defines the full lists of Dynamic Functions for Leaf Nodes.

Elementary Functions

Stochastic (R): 	Constant (C):
Increase (I): 	Decrease (D):

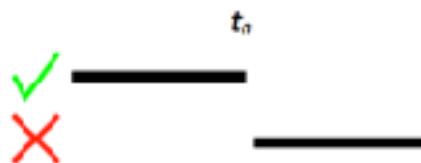
Denied-Satisfied (DS)

Patterns:



Satisfied-Denied (SD)

Patterns:



Monotonic Negative (MN)

Patterns:



Monotonic Positive (MP)

Patterns:



Note: FS is Satisfied (or Fully Satisfied) and FD is Denied (or Fully Denied)

Elementary Functions	
<i>Constant (C)</i>	the satisfaction evaluation remains constant at <i>constantValue</i>
<i>Increase (I)</i>	changes in satisfaction evaluation become “more true” to a <i>maxValue</i> as time progresses
<i>Decrease (D)</i>	changes in satisfaction evaluation become “less true” to a <i>minValue</i> as time progresses
<i>Stochastic (R)</i>	changes in satisfaction evaluation are stochastic or random
General Compound Function	
<i>User-Defined (UD)</i>	its value is a stepwise function defined by a sequence of other functions, repeating behaviour can be specified over a subset of the function
Common Compound Functions	
<i>Satisfied-Denied (SD)</i>	the satisfaction evaluation remains <i>FS</i> until t_i and then remains <i>FD</i>
<i>Denied-Satisfied (DS)</i>	the satisfaction evaluation remains <i>FD</i> until t_i and then remains <i>FS</i>
<i>Stochastic-Constant (RC)</i>	changes in satisfaction evaluation are stochastic or random until t_i and then remains constant at <i>constantValue</i>
<i>Constant-Stochastic (CR)</i>	the satisfaction evaluation remains constant at <i>constantValue</i> until t_i and then changes in evaluation are stochastic or random
<i>Monotonic Positive (MP)</i>	changes in satisfaction evaluation become “more true” to a <i>maxValue</i> at t_i and then remains constant at <i>constantValue</i>
<i>Monotonic Negative (MN)</i>	changes in satisfaction evaluation become “less true” to a <i>minValue</i> at t_i and then remains constant at <i>constantValue</i>

(Optional) GrowingLeaf Tool: Quick Instructions

GrowingLeaf allows you to draw i* models using the shapes from the Palette on the left. Click the desired element or actor from the left palette and drag in onto the canvas. Release mouse.

How to name an actor or element

Click on the element you want to rename on the canvas. On the top right of the screen appears a Node Name box, initially with the general name, e.g., Task. Replace this text with the desired name then click back on the canvas.

How to draw a link

Click on the element that will be the source of the link. A black arrow appears on the right side of the element. Click and drag this arrow to another shape, the destination of the link. When you are close enough to the other element, a red box will appear around it. Release the mouse button and the link is connected from source to target.

How to pick a link type

By default, all new links are And-decomposition. To change a link type, hover over a link. A black gear icon appears. Click on the icon. Now in the top right where you have previously been able to name an element the Link type drop-down box appears. Pick the type of link you want from the drop-down box.

Hint: if the gear symbol is too small or overlaps with other icons, try moving the element slightly.

How to save/load a file

Use the save and load buttons on the top of the screen. They prompt you for a file name and location. By default the file extension added is .json