Getting Started with AgentSheets

Thought Amplifier.

AgentSheets®

Contents

Trademarks and Copyrights

Install AgentSheets

System Requirements

AgentSheets runs efficiently on Pentium II or Celeron computers and runs still acceptably on older Pentium PCs. The system requirements of AgentSheets 2.1 for the Microsoft Windows Platforms are:

◦	**Minimum**	**Recommended**
Free hard disk space:		
AgentSheets + JRE[1]	30 MB	30 MB
Documentation	5 MB	5 MB
Tutorial Videos	200 MB	200 MB
System RAM	64MB	96MB
CPU speed	200 MHz	300 MHz

Installation

The AgentSheets CD contains a pre-installed version of the AgentSheets application including documentation. To install AgentSheets onto a PC copy this pre-installed version in the AgentSheets 2.1 folder. Alternatively, you can run the **asj2_1.exe** installer in the AgentSheets Installer folder.

asj2_1exe

Launch

Launch AgentSheets by double-clicking the AgentSheets application on your desktop, or by going to `StartMenu | Program files| Agentsheets2.1` and single clicking on the icon there.

AgentSheets 2.1

Registration

For full AgentSheets functionality, you need your registration number, which is provided separately. On-line buyers receive the registration number through email.

---

[1] JRE = Java Runtime Environment

Introduction

The purpose of the *AgentSheets*¤ *Getting Started Guide* is to provide you with step-by-step instructions for creating an AgentSheets simulation. It is assumed that you know the basics of using a computer running Microsoft Windows, but no programming experience is required.

What is AgentSheets?

AgentSheets is an agent-based simulation-authoring tool that lets a wide range of end users (from children to professionals) create their own interactive simulations and games. AgentSheets is a revolutionary environment that combines *agents*, *spreadsheets* and *Java authoring* technologies in a single medium. AgentSheets acts like a thought amplifier that lets you build on your own understanding, and explore new ideas. The Java technology —called Ristretto¤ — built into AgentSheets lets you communicate your ideas through the Web to the rest of the world.

To get started, you only need an open mind and an active imagination. AgentSheets lets individual creativity flourish in a programming environment that is both intuitive and transparent.

What is an Agent?

In AgentSheets, agents are end-user programmable objects. Agents react to mouse clicks and keyboard input, move around, change their appearance, play MIDI music and videos, speak, read Web pages, send email, and compute formulae. Agents don t operate alone. Dozens, hundreds or even thousands of agents interact with each other in a spreadsheet-like grid to create an AgentSheets simulation.

How was AgentSheets Born?

The AgentSheets programming environment is the product of ten years of research. Initially, AgentSheets grew out of the idea of building a new kind of computational media that lets computer users build highly parallel and interactive simulations by replacing spreadsheet numbers and strings with behaving agents. In this initial stage, AgentSheets was an intricate simulator that required powerful machines and professional-level programming. Over time,

AgentSheets evolved into a fast simulator that runs on personal computers and requires no programming experience. This evolution was made possible by a completely different approach to programming. We call this new approach "tactile programming."

Tactile Programming

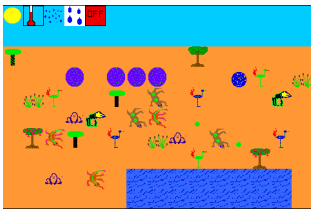
Tactile Programming is the modern foundation of AgentSheets. Tactile Programming moves one conceptual step beyond Visual Programming. While most programming languages have been developed from a technological perspective, the design of the AgentSheets environment has been driven by people's common need to visualize, understand and communicate ideas. Of these, the communication of ideas is perhaps the most important; AgentSheets excels at this function.

Tactile Programming allows you to literally experience programming by manipulation. The AgentSheets programming language, called Visual AgenTalk¤ (VAT), is a rule-based language that features conditions, actions and rules. *Conditions*, *actions* and *rules* are complete objects that you explore. At any time, you select any condition, action or rule and test it without first having to construct a complete program. Explore questions like: is this condition true for this agent; what will happen if this agent runs this action; will this rule fire, and if so what will be the consequences? Moreover, you can select any condition, action and rule and have AgentSheets explain it through animated tool tips. AgentSheets promotes a more playful attitude towards programming, one that thrives on immediate feedback and promotes exploration.

AgentSheets brings the "art of programming" to artists and content developers who might otherwise be constricted by the daunting syntax of languages like C++ or Java. With AgentSheets, anyone can translate their ideas into a computer program.

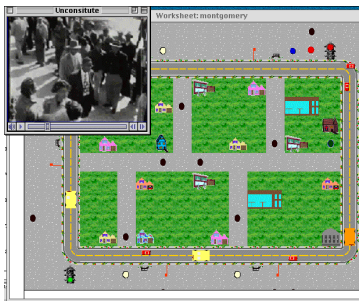
Examples

AgentSheets has been used by a wide range of end users (from children to professionals) to create their own interactive simulations and games. Here are some examples:



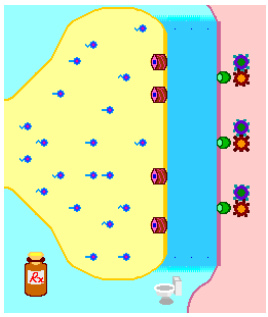
K-12 Education: Elementary School

Collaborative Learning: Students learn about life science topics such as food webs and ecosystems by designing their own animals. Collaborative animal design takes place when groups of students put their individual animals into shared worlds to study the fragility of their ecosystems.



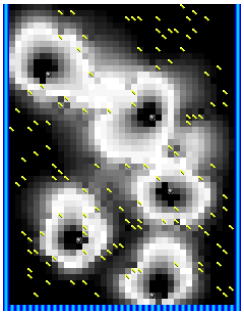
K-12 Education: High School

Interactive Story Telling: History students create interactive stories of historical events such as the Montgomery bus boycott.



Training

Distance Learning: With SimProzac, patients can explore the relationships among Prozac, the neurotransmitter serotonin, and neurons. By playing with this simulation in their browsers, patients get a better sense of what Prozac does than they ever could by reading the cryptic description included with the drug.

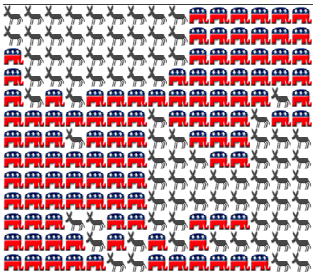


Scientific Modelling

Learning by Visualization and Modeling: Researchers model the effects of microgravity on E.coli bacteria. This is an AgentSheets simulation of an experiment that flew on the Space Shuttle with John Glenn. This particular simulation uses thousands of agents.

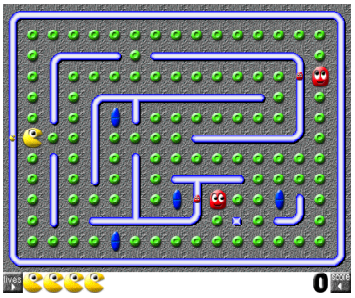
Educational Games

Learning Through Simulation Use: This simple voting simulation explains concepts such as clustering, migration and the stability of two party systems. Can it predict the outcome of the next election?



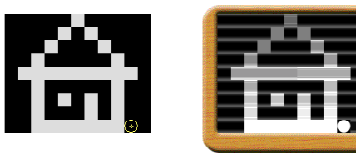
Non-Educational Games

Learning Through Design: Even if the finished simulation/game is not directly related to education, the construction process is very educational. The Ultimate Pacman is a complete game based on complex Artificial Intelligence algorithms and non-trivial diffusion equations.



Interactive Illustrations

How Does a TV Work? This simulation illustrates how a picture is scanned by a camera (left), transmitted to a TV set and then converted back to a picture (right). Users can paint their own pictures and play with TV signal processing parameters.



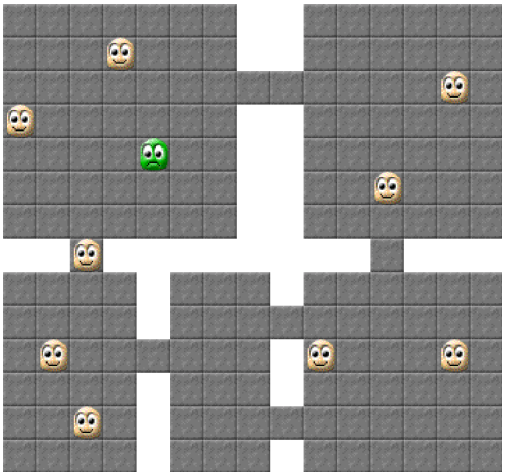
Deconstruction Kits

Learning by Taking Apart: What makes a bridge stable? The task presented to users by this simulation is to remove as many bridge elements as possible without collapsing the bridge. Implicit learning experiences include forces, architecture, and geometric perspective. This simulation was featured on PBS Mathline.

Tutorial: Virus Attack

To illustrate how an AgentSheets simulation is created from scratch, we present an 8-step description on how to build *Virus Attack*, a simulation of a virus spreading through a community. If you prefer to follow a movie, run the Getting-Started movie on the AgentSheets CD. In less than an hour you will have built a complete simulation and turned it into a Java applet ready to be published on the Web.

The point of Virus Attack is to understand the basic virus-spreading mechanism. Does the number of people infected by the virus increase linearly, or is there some other function behind the rate of growth? How fast does the virus spread? What can be done to contain the virus? The goal is to explore issues associated with the spread of a virus by creating, running, and studying a simulation. The principles behind this kind of simulation are not limited to virus propagation. The same laws apply to many other things that spread socially, such as rumors or fads.



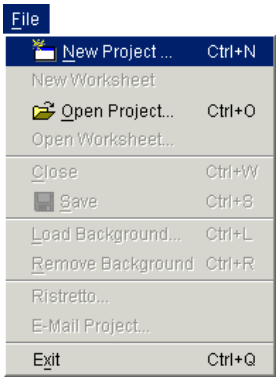
In Virus Attack, simulated people (called *agents* in AgentSheets) move around randomly to represent the real activities of people going to work, shopping and traveling. A healthy person standing next to an infected person has a 5% chance of getting the virus. A completed version of this project is shown above.

1. Create a Project

In AgentSheets, a project consists of the set of agents which are organized in a *gallery*, the agents' behavior, and the simulation worlds (called *worksheets*) in which agents interact.

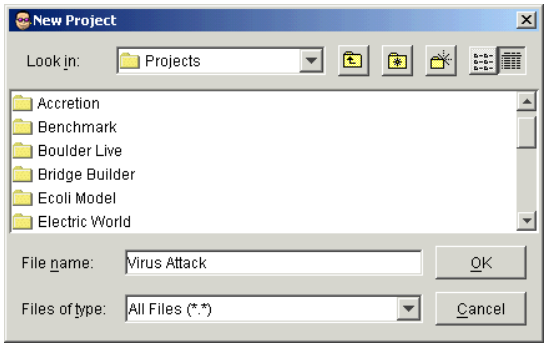
Create a New Project

Create a new project with the `File| New Project...` menu option shown in the margin.



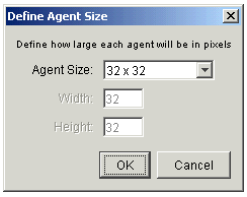
Save the Project as a Folder

The `New Project` dialog will appear, asking you to specify the name of your new project and the location where you wish to save your project. Type `Virus Attack` as the name of your new project (in the File name field) and click `OK`. The default location to save new projects is the Projects folder of the AgentSheets application.



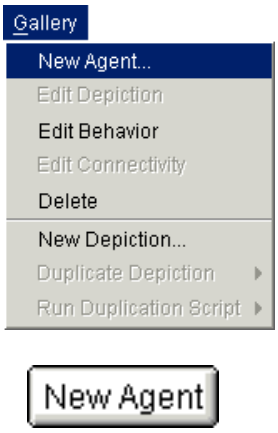
Define the Size of Your Agent

A dialog box will ask you to specify the size (in pixels) of the agents for this project. The optimal size for an agent depends on what you are simulating. For the Virus Attack, we will use a custom agent size of 24 x 24 pixels. An empty `Gallery` window is created for you.



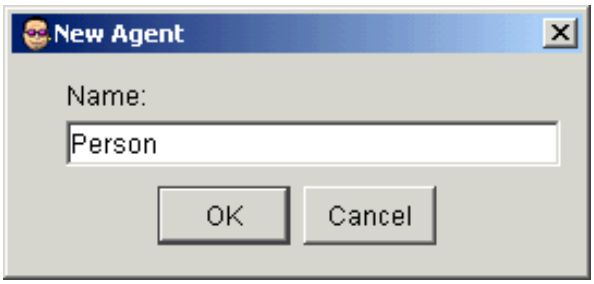
Each project has one gallery, which is used to organize all agents that you are about to create.

2. Define Agents



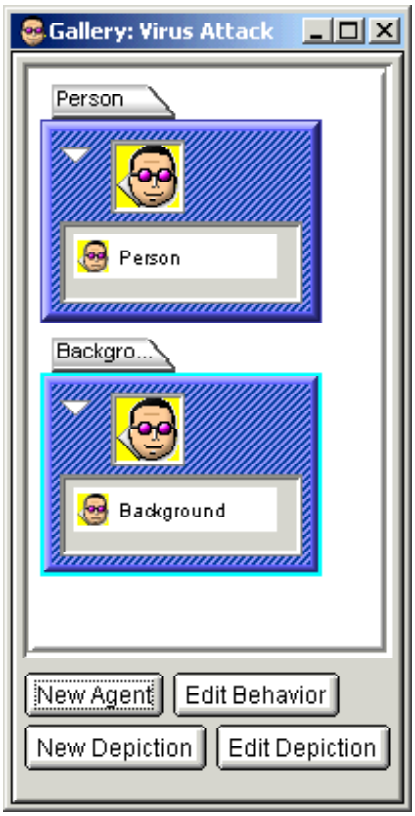
Agents represent all the objects of your simulation. For the Virus Attack project, you need agents to represent healthy people, infected people, and background tiles on which the agents move.

Choose the `Gallery| New Agent..` menu option or click the `New Agent` button in the gallery to create a new agent. A dialog box will ask you to name your agent. Name the agent Person and click `OK`.



Use `Gallery| New Agent..` or the `New Agent` button again to create another agent called Background .

Each of these agents will show up in the gallery as a box labeled with the name you just specified. An agent icon is provided by default. Next, we will create an appearance for each agent.

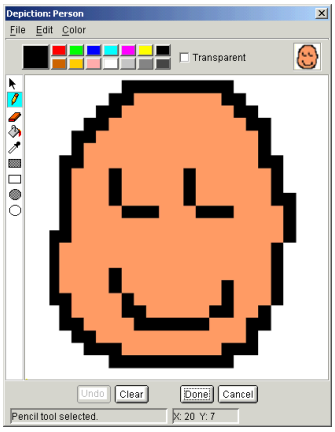


3. Edit Agent Depiction

An agent can have one or more depictions defining how the agent will look on the screen. If the list of depictions associated with the agent is not visible in the gallery, click on the disclosure triangle on the agent and it will appear.

Edit Person and Background Agent Depictions

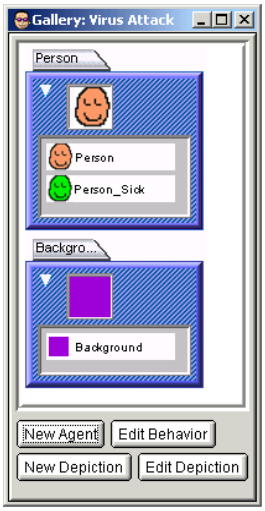
To edit an agent s depiction, use the AgentSheets *depiction editor*. To get the depiction editor, double-click the default depiction of **Person** (outlined in red) or select the depiction and click the `Edit Depiction` button on the gallery. Clear the default depiction using the `Clear` button in the depiction editor. Use the **Pencil** tool and various ink colors to draw an icon resembling the face of a person.



Once you are pleased with your creation, click the `Done` button, which will save the depiction and close the depiction editor.

Double-click the **Background** agent's default depiction in the gallery and edit its depiction to look like a large floor tile. You can just select a color and use the **Fill** tool to fill the entire square. Close the depiction editor and save the **Background** depiction.

Create a Person_Sick Depiction



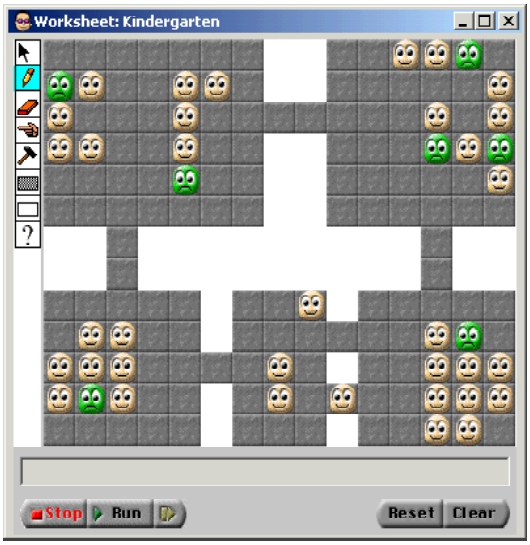
We need a way to show that somebody has acquired the virus. A simple strategy is to create a new depiction for the **Person** agent representing a sick person. Select the **Person** agent in the gallery. Create a new depiction by using the `Gallery| New Depiction…` menu option or by clicking on the `New Depiction` button at the bottom of the gallery and name the depiction Person_Sick". Double-click this new depiction and use the depiction editor **Fill** tool with a different ink color to show a sick person.

4. Create a Simulation World

Now you are ready to define a simulation world (a worksheet), where the Virus Attack simulation will take place. Use the `File | New Worksheet` menu option to create a new worksheet.

To draw **Background** agents in the worksheet, select the **Background** depiction in the gallery. Select the **Draw Rectangle** tool on the left of the worksheet**.** Click and drag to outline the region of the worksheet you want filled with **Background** agents, then release the mouse.

The simulation is more interesting if you don t just draw one big area of **Background** agents. Instead, draw different sized, connected blocks of **Background** agents. Think of them as rooms and corridors.



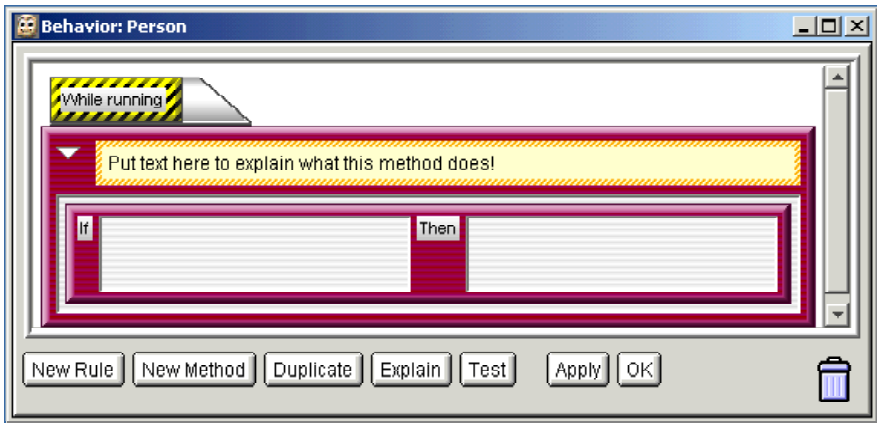
To add people, select the **Person** depiction in the gallery. Select the **Pencil** tool and drop several **Person** agents into the worksheet. Then, select the **Person_Sick** depiction in the gallery. Now you can add one **Person** agent with the **Person_Sick** depiction into the worksheet to start the epidemic. Save your worksheet to a file, by selecting it, choosing `File| Save`and then naming it Kindergarten . (Please note that we are now using higher quality depictions, which replaced the ones created in the previous section.)

If you run the simulation, since no behaviors are defined, the scene is completely passive at this point.

5. Open a Behavior Editor

In the next two steps, we will construct complete agent behaviors by combining conditions and actions into rule*s* and by grouping rules into methods in a *behavior editor*. A *behavior* describes what an agent does. Agent behaviors are expressed in Visual AgenTalk as IF-THEN rules containing actions and conditions. Actions and conditions are the building blocks used to define an agent's behavior.

Double clicking the blue texture area of an agent box in the gallery or selecting the agent and clicking the `Edit Behavior` button opens the behavior editor for that agent.

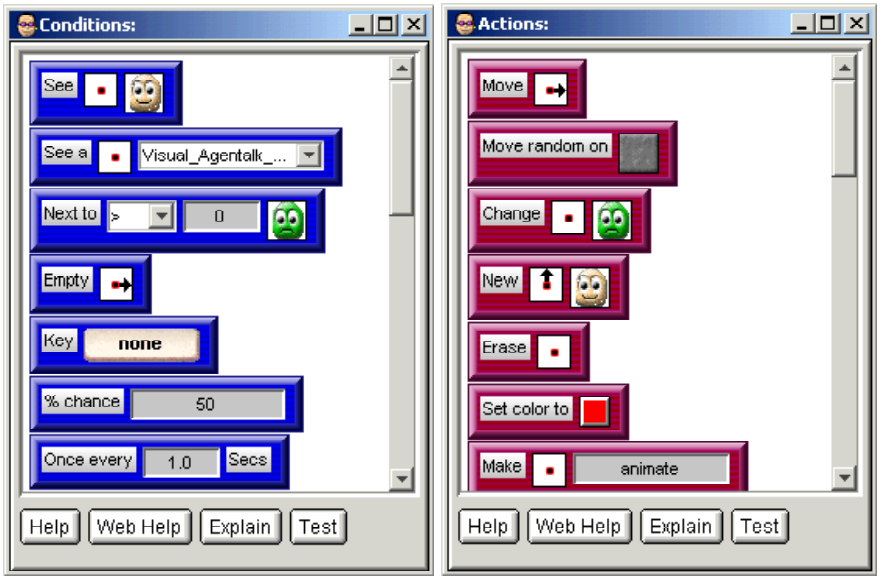


A behavior editor can contain any number of method*s*. A *method* contains a list of rules and a trigger. The *trigger* defines when the rules contained in the method get checked. The behavior editor above contains one method labeled with the **While Running** trigger**.** The **While Running** trigger will make the method check its rules once per simulation cycle.

6. Define Rules

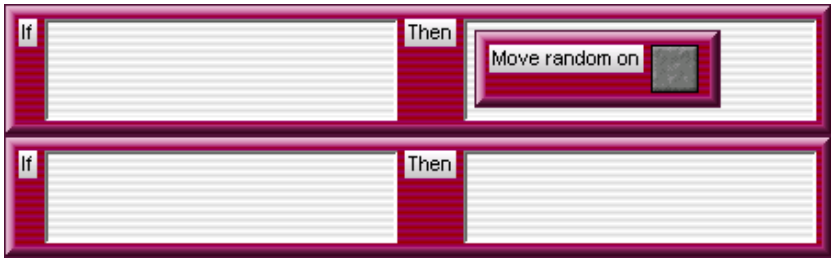
To simulate spreading a virus, we need a vulnerable **Person** agent that runs around randomly on the **Background**. We can make the agent vulnerable by giving it some chance of becoming infected if it encounters an infected **Person** agent. This kind of behavior can be expressed in rules. We will now select conditions from the conditions palette and actions from the actions palette and drag them into our behavior editor to construct the desired behavior.

Open the `Conditions:` palette via the `Tools| Conditions Palette` menu option or by clicking the blue conditions palette button (shown in the margin). Open the `Actions` :palette via the `Tools | Actions Palette` menu option or by clicking the red actions palette button (shown in the margin).



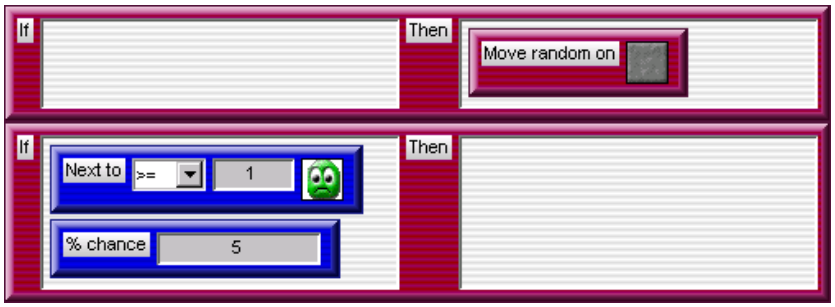
To make the **Person** agent move randomly on the **Background**, select the **Move Random On** action from the `Actions` :palette and drag it into the THEN Part of the **Person** agent s behavior editor. Make sure you select the **Background** depiction as a parameter to the **Move Random On** action. Your first rule is complete.

Let us define a second rule to spread the virus. Click the `New Rule` button at the bottom of the behavior editor to create a new rule.

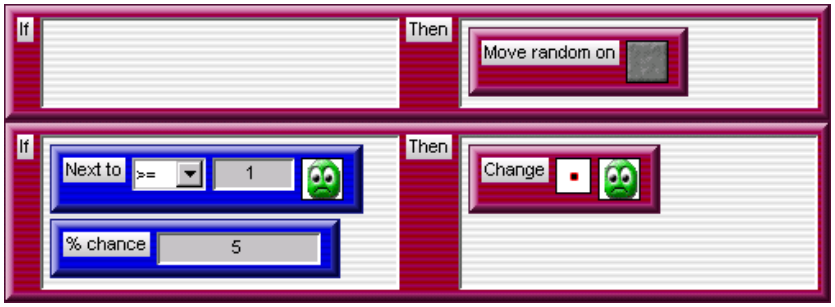


A **Person** agent can get infected if it is next to at least one person carrying the virus. Drag a **Next To** condition from the `Conditions:` palette into the IF part of the second rule. Change the test in the **Next To** condition to >= , change the number of agents required to 1, and select the **Person_Sick** depiction.

The virus should only spread with a 5% chance. Drag a **% Chance** condition into the same IF box below the **Next To** condition. Define the probability in the **% Chance** condition to 5.



If the agent is next to an infected agent, there is a 5% chance that the agent will also get infected. To achieve that, drag a **Change** action into the second rule s THEN part and select the **Person_Sick** depiction as a parameter.



Rearrange the Rules

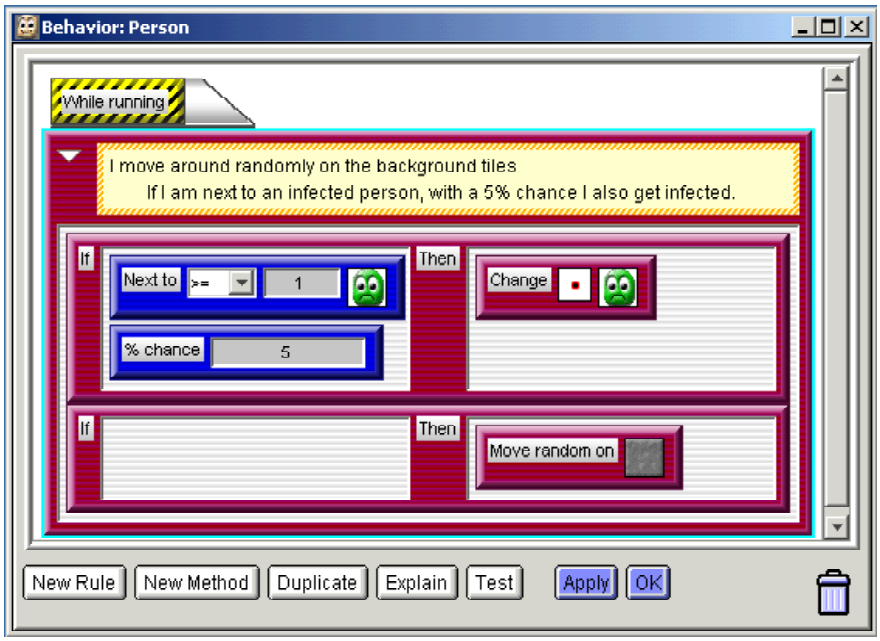
AgentSheets checks an agent s rules as follows:

¥ ***Check*** the list of rules starting at the top working towards the bottom.

¥ ***Fire*** the first rule whose conditions are all true. All actions of that rule will be executed. Rules below the fired rule will not be checked.

Given this, we have a problem with our current rule arrangement. Our rule at the top of the list has no condition and will fire immediately without giving the second rule a chance to fire. Therefore, our **Person** agent would always move randomly on the **Background** and would never check if there were infected agents around it.

Fortunately, there is a simple fix for our problem. Change the order of the rules by grabbing the second rule from the Then label and dragging it above the first rule. Now the agent will start by checking for infected agents in its vicinity. It will move only if this is not the case.



7. Run the Simulation

Once you are ready to test your agent's behavior, apply your newly created rules by clicking on the `Apply` button of the behavior editor. To start the Virus Attack simulation, press the `Run` button in the worksheet. You can then observe how the virus spreads. You can add and remove agents while the simulation is running.

If the simulation is running too fast for you to follow, stop it and use the `Step` button to run the simulation one step at a time. The `Step` button is to the right of the `Run` button.

Note that the `Apply` button makes the behavior specified in the editor take effect for all instances of the agent and saves changes to the agent's behavior file. The `OK` button also makes the changed behavior take effect, but in addition it closes the editor.

The blue color of the `Apply` and `OK` buttons are indicators that the behavior of an agent was changed since the last time you hit either the `Apply` or `OK` button.

8. Ristretto: Create a Java Applet

You have created a complete interactive simulation that runs locally on your computer. How can you share your simulation? AgentSheets includes the unique Ristretto Technology that lets you instantly create a Java applet.

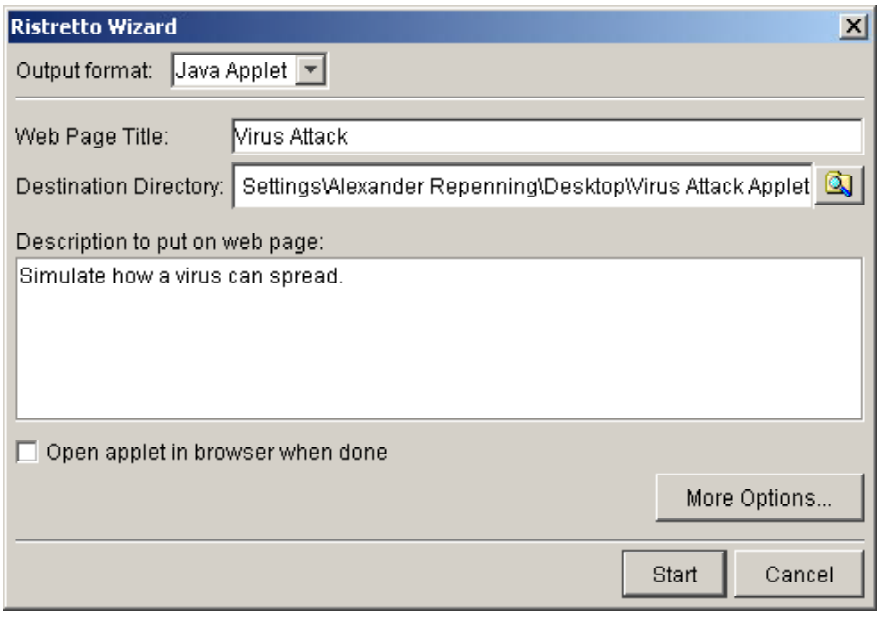
What is a Java Applet?

A Java Applet is a cross-platform program that will run in a Web page. This means that anyone with a Java-enabled Web browser can run an applet on Mac OS, Windows, Unix, Linux or any other operating system. Once you turn an AgentSheets project into a Java applet, you may copy it onto a Web server, a floppy, a CD or any other storage media. You can even email the applet. No other simulation-authoring tool allows this without plugs-in or players.

If you run applets on a Macintosh, we recommend that you use a Web browser that directly supports Apple's Java Runtime Environment, MRJ.

Create Your Applet with the Ristretto Button

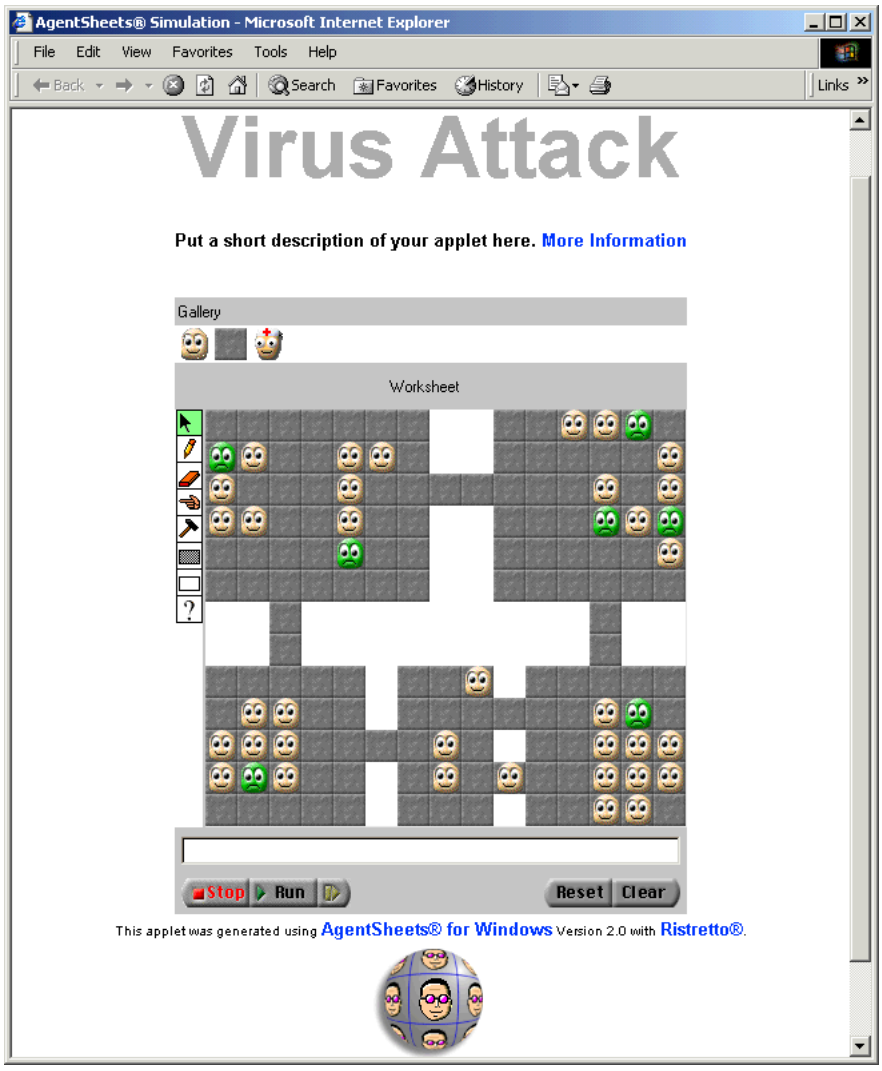
Creating a complete Web page that includes your Java applet is simple: just press the `Ristretto`button. Specify a folder where your applet will be saved (the default location is the desktop) and press the `Start`button.



It is good practice to fill out the Description to put on web page field. You can use this description to explain what your simulation is about and to instruct users on how to use your simulation. Press the More Options button to control the appearance of your applet. A status bar indicates progress while AgentSheets automatically generates a complete Web page that includes your applet.

Run your Applet

Locate the folder in which you saved your applet. Inside this folder you will find an index.html file. Double click this file and you should see your applet appear in your Web browser. If you do not have a Java-enabled Web browser, we recommend downloading Internet Explorer at www.microsoft.com.



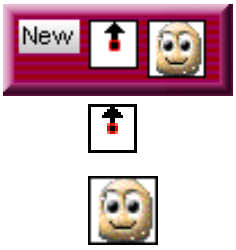
The simulation is finished and running in a Web page. Congratulations!

Getting More Help

Creating agents' behaviors would usually be the hard part, but fortunately AgentSheets helps you gradually develop complex agent behavior and learn as you go. Mechanisms such as the Test and Explain buttons enable users to explore the Visual AgenTalk language.

Explore Actions

Actions, which can be displayed with the `Tools | Actions Palette` menu option, are operations performable by agents. Actions allow agents to do things such as moving around a worksheet, changing their depiction, playing a sound, or opening a Web page.



Actions may include parameters, such as the **Direction** parameter and the **Depiction** parameter, which can be directly manipulated. Click the **Direction** parameter, for example, to specify a direction in which the new agent will be created in the **New** action and click the **Depiction** parameter to select what the new agent will look like.

Test Actions

Each action is a tactile object that can be directly manipulated. You can explore an action by selecting it in the actions palette, selecting a specific agent in the worksheet using the Arrow tool, and clicking the `Test` button. Testing the **Move** action onto a **Person** agent in the worksheet will make the agent move in the direction indicated by the arrow in the **Direction** parameter. Any action can be tested on any agent in the worksheet. You can do this anytime you are curious about an action.

Testing an action on an agent will make the agent execute the command once. The action s consequences also depend on the context. For example, when tested on the healthy **Person** agent, the **Change** action executes, turning the healthy **Person** agent into a sick **Person** (that is a **Person** agent with the **Person_Sick** depiction). This mechanism lets you test any action to see what it does. An action that is currently executing provides visual feedback by changing its frame to a yellow and black striped pattern. This is especially useful when commands are embedded in programs with multiple commands, rules, and methods.

Explain Actions

If you want an explanation of what an action does, select the action in the actions palette and press the `Explain`button. A context sensitive explanation of the action is provided in an animated tool tip window below the action.

The system steps through the parts of the action being explained, while the corresponding piece of explanation gets highlighted.

Access Action Documentation

For more detailed help, select an action and click the `Web Help` button at the bottom of the actions palette. This brings up a web browser presenting information relevant to the selected action. Web Help indexes directly into the AgentSheets Reference Manual.

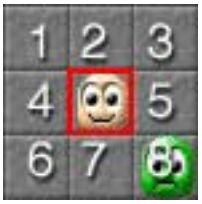
Explore Conditions

Conditions, which can be displayed with the `Tools | Conditions Palette`menu option, are used to test an agent s circumstances. Conditions are either true or false. Among other things, agents can use conditions to test for the presence of other agents around them, test attribute values, detect keyboard input and mouse events, and even search content from live Web pages.

Test Conditions

Any condition can be tested by selecting it in the conditions palette, selecting an agent in the worksheet using the Arrow tool, and clicking the `Test` button. As usual with AgentSheets, you do not have to create a complete program to test conditions. This lets you "play" with conditions to find out what they do and where they apply.

For example, testing the **Next To** condition on an agent will test if any of the 8 adjacent agents looks like the depiction specified in the condition.



A yellow and black striped frame around the condition indicates that the condition is being tested. If the condition is true then the

frame quickly disappears and a sound is played. If the condition is false, the frame begins to blink and a different alert sound is played. In the situation above, the condition is true since there is one or more adjacent agents that looks like the one specified by the condition.

Explain Conditions

If you want an explanation of what a condition does, select the condition in the conditions palette and press the `Explain`button. A context sensitive explanation of the condition is provided in a tool tip window below the condition.

The system steps through the parts of the condition being explained, while the corresponding piece of explanation gets highlighted.

Access Condition Documentation

For more detailed help, select a condition and click the `Web Help` button at the bottom of the conditions palette. This brings up a web browser presenting information relevant to the selected condition. Web Help indexes directly into the AgentSheets Reference Manual.

Explore Rules

Rules are sets of conditions and actions combined in IF-THEN structures.

Test Rules

At any point during the creation of a simulation, rules can be tested by running the simulation. Also, entire rules can be selected in the behavior editor and tested in specific contexts by clicking the `Test` button. If the rule can fire - meaning that all its conditions are true - it will execute all its actions. If the rule cannot fire, it will indicate why it cannot fire by making the unsatisfied condition blink.

Explain Rules

Select any rule in the behavior editor by clicking its Then label. Press the `Explain`button. A context sensitive explanation is provided by stepping through the entire rule and having its components be explained in an animated tool tip window below each command of the rule.

Explore Methods

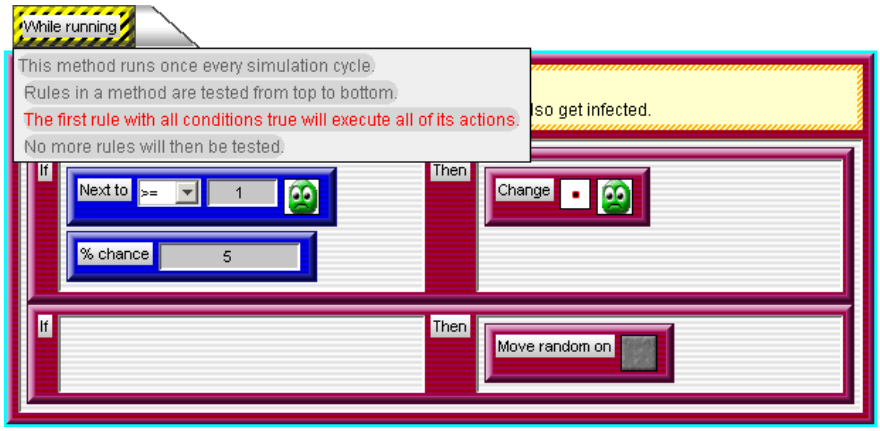
Lists of rules are grouped together into *methods*. Each method is labeled with a *trigger*. A full list of triggers is housed in the `Triggers` palette which is accessed via the `Tools | Triggers Palette` menu or by clicking the Trigger palette icon.

Test Methods

You can test an entire method by selecting both the method and the agent you want to test it on in the worksheet and then clicking on the `Test` button. The system will step through all the rules, giving you feedback on what is being executed at each step. When the system finds a rule whose conditions are all true, it steps through the list of actions specified in that rule.

Explain Methods

Select any method in the behavior editor by clicking its trigger or its tab. Press the `Explain`button. An explanation for the selected method is provided in a tool tip window below the method s trigger.



Other Resources

You can get find more helpful information about the AgentSheets software in the following documents, found in the Documentation folder on the AgentSheets CD:

- **Language Reference Manual**: A detailed description of all the Visual AgenTalk language elements (conditions, actions, triggers and parameters) including example simulations that use them. Format: printed manual, PDF, HTML.
- **Reference Manual**: A description of the complete menu structure of AgentSheets including all the dialog boxes, tools, action commands, condition commands, triggers, parameters and simulation properties. Format: PDF, HTML.
- **Anatomy of a Project**: A dissection of the AgentSheets project folder structure with explanations of each component. Format: PDF.

Acknowledgements

This work was paid for in part by funding from the National Science Foundation.

This material is based upon work supported by the National Science Foundation under award numbers DMI 9761360, REC 9804930, REC-9631396, RED 9253425, CDA-940860. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Software License and Limited Warranty

AGENTSHEETS, INC.

End-User License Agreement

1 User

NOTICE TO USER:

THIS IS AN AGREEMENT. PLEASE READ IT CAREFULLY. BY INSTALLING THIS SOFTWARE YOU ACCEPT ALL THE TERMS AND CONDITIONS OF THIS AGREEMENT. IF YOU DO NOT AGREE WITH THE TERMS AND CONDITIONS OF THIS AGREEMENT, YOU MUST NOT USE THE SOFTWARE. IF YOU DO NOT ACCEPT THIS AGREEMENT AND HAVE PROOF OF PAYMENT, YOU MAY RETURN THE UNUSED SOFTWARE TO THE LOCATION AT WHICH YOU ACQUIRED IT WITHIN THIRTY (30) DAYS FOR A REFUND OF THE LICENSE FEE OR IF PURCHASED DIRECTLY FROM AGENTSHEETS, INC., YOU SHOULD RETURN IT TO AGENTSHEETS, INC. AT 6560 GUNPARK DR., SUITE D, BOULDER, CO 80301 WITH PROOF OF PAYMENT WITHIN THIRTY (30) DAYS FOR A REFUND OF THE LICENSE FEE.

1. **USE OF THE SOFTWARE**.

You may install one copy of the Software onto a hard disk or other storage device up to the Permitted Number of Computers.

You may install one copy of the Software on a single file server for the purpose of downloading and installing the Software onto a hard disk or other storage device of up to the Permitted Number of Computers that are on the same network as the file server. No other network use is permitted.

You may make one backup copy of the Software, provided your backup copy is not installed or used on any computer.

HOME USE. The primary user of each computer on which the Software is installed may also install the Software on one home computer. However, the Software may not be used on the home computer at the same time the Software on the primary computer is being used.

APPLICATIONS. Unless stated otherwise in the Documentation, you may display, modify, reproduce, and distribute any of the Applications included with the Software. However, you may not distribute the Applications on a stand-alone basis, i.e., in circumstances in which the Stock Files constitute the primary value of the product being distributed. You should review the “Read Me” files associated with the Applications that you use to ascertain what rights you have with respect to such materials. You may not register or claim any trademark rights in the Applications or derivative works thereof.

APPLETS AND SIMULATIONS. Applets and Simulations may not be used in the production of libelous, defamatory, fraudulent, infringing, lewd, obscene, or pornographic material or in any otherwise illegal manner.

DISTRIBUTION OF CONTENT. End-user generated projects converted through the use of the Ristretto tool into Java applets can be distributed, permitted that the AgentSheets marks and links to the AgentSheets web site are left intact. The marks include references to "AgentSheets" and "Ristretto" as well as the AgentSheets logo. If these marks are removed you need to contact AgentSheets, Inc., 6560 Gunpark Dr., Suite D, Boulder, CO 80301 to negotiate a royalty-based license agreement.

2. **COPYRIGHT**. The Software and any copies that you make are owned by AgentSheets Inc., and its structure, organization, and code are the valuable trade secrets of AgentSheets. The Software is also protected by Copyright Law and International Treaty provisions. You must treat the Software just as you would any other copyrighted material, such as a book. You may not copy the Software, except as set forth in the “Use of the Software” section. Any copies that you are permitted to make pursuant to this Agreement must contain the same copyright and other proprietary notices that appear on or in the Software. You agree not to modify, adapt, or translate the Software. You also agree not to reverse engineer, decompile, disassemble, or otherwise attempt to discover the source code of the Software. Trademarks shall be used in accordance with accepted trademark practice, including identifications of trademark owners’ names. Trademarks can only be used to identify printed output produced by the Software and such use does not give you any right of ownership in that trademark. Except as stated above, this Agreement does not grant you any intellectual property rights in the Software. This Agreement provides the terms and conditions under which you are licensed to use the Software. It is not an agreement for the sale of the Software to you, and ownership of the Software shall be and remain in AgentSheets, Inc.

3. **TRANSFER.** You may not rent, lease, sublicense, or lend the Software. You may, however, transfer all of your rights to use the Software to another person or legal entity provided (1) that you transfer this Agreement, the Software, including all copies, updates, and prior versions, to such person or entity, (2) that you retain no copies, including copies stored on a computer, and (3) that the receiving party accepts the terms and conditions of this Agreement and such terms and conditions shall apply to and be binding upon such receiving party.

4. **MULTIPLE ENVIRONMENT SOFTWARE/MULTIPLE LANGUAGE SOFTWARE/DUAL MEDIA SOFTWARE/MULTIPLE COPIES/UPDATES**. If the Software supports multiple platforms or languages, you receive the Software on multiple media, or you otherwise receive multiple copies of the Software, the number of computers on which all versions of the Software are installed may not exceed the Permitted Number of Computers. You may not rent, lease, sublicense, lend, or transfer versions or copies of the Software you do not use. If the Software is an Update to a previous version of the Software, you must possess a valid license to such previous version to use the Update, and you may use the previous version for ninety (90) days after you receive the Update to assist you in the transition to the Update. After such

time you no longer have a license to use the previous version, except for the sole purpose of enabling you to install the Update.

5. **LIMITED WARRANTY.** AgentSheets warrants to you that the Software will perform substantially in accordance with the Documentation for the ninety (90) day period following your receipt of the Software. To make a warranty claim, you must return the Software to the location where you obtained it along with proof of purchase with such ninety (90) day period. If the Software does not perform substantially in accordance with the Documentation, the entire liability of AgentSheets and your exclusive remedy shall be limited to either, at AgentSheets's option, the replacement of the Software or the refund of the license fee you paid for the Software. THE FOREGOING STATES THE SOLE AND EXCLUSIVE REMEDIES FOR AGENTSHEETS' BREACH OF WARRANTY. THE LIMITED WARRANTY SET FORTH IN THIS SECTION GIVES YOU SPECIFIC LEGAL RIGHTS. YOU MAY HAVE OTHER RIGHTS, WHICH VARY FROM STATE TO STATE OR JURISDICTION TO JURISDICTION. For further warranty information, please contact AgentSheets' Customer Support Department.

6. **DISCLAIMER OF WARRANTIES**. EXCEPT FOR THE LIMITED WARRANTY SET FORTH IN SECTION 5, AGENTSHEETS MAKES NO WARRANTIES, EXPRESS, IMPLIED, OR STATUTORY, AS TO ANY OTHER MATTERS, INCLUDING BUT NOT LIMITED TO NON-INFRINGEMENT OF THIRD-PARTY RIGHTS, MERCHANTABILITY, OR FITNESS FOR ANY PARTICULAR PURPOSE. AGENTSHEETS DOES NOT AND CANNOT WARRANT THE PERFORMANCE OR RESULTS YOU MAY OBTAIN BY USING THE SOFTWARE. Some states or jurisdictions do not allow the exclusion of implied warranties or limitations on how long an implied warranty may last, so the above limitations may not apply to you. To the extent permissible, any implied warranties are limited to ninety (90) days.

7. **LIMITATION OF LIABILITY**. IN NO EVENT WILL AGENTSHEETS BE LIABLE TO YOU FOR ANY CONSEQUENTIAL, INDIRECT, INCIDENTAL, PUNITIVE, OR SPECIAL DAMAGES, INCLUDING ANY LOST PROFITS OR LOST SAVINGS, EVEN IF A REPRESENTATIVE OF AGENTSHEETS HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, OR FOR ANY CLAIM BY ANY THIRD PARTY. SOME STATES OR JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THE ABOVE LIMITATIONS MAY NOT APPLY TO YOU.

8. **GOVERNING LAW AND GENERAL PROVISIONS**. This Agreement will be governed by the laws in force in the State of Colorado excluding the application of its conflicts of law rules. This Agreement will not be governed by the United Nations Convention on Contracts for the International Sale of Goods, the application of which is expressly excluded. If any part of this Agreement is found void and unenforceable, it will not affect the validity of the balance of the Agreement, which shall remain valid and enforceable according to its terms. You agree that the Software will not be shipped, transferred, or exported into any country or used in any manner prohibited by the United States Export Administration Act or any other export laws, restrictions, or regulations. This Agreement shall automatically terminate upon failure by you to comply with its terms, in which event you must destroy all copies of the Software. This Agreement may only be modified by a writing signed by an authorized officer of AgentSheets, although AgentSheets may vary the terms of this Agreement in connection with the licensing of any Updates to you.

9. **NOTICE TO U.S. GOVERNMENT END USERS**. The Software and Documentation are "Commercial Items," as that term is defined at 48 C.F.R. § 2.101, consisting of "Commercial Computer Software" and "Commercial Computer Software Documentation," as such terms are used in 48 CFR. § 12.212 or 48 C.F.R. § 227.7202, as applicable. Consistent with 48 C.F.R. § 12.212 or 48 C.F.R. §§ 227.7202-4, as applicable, the Commercial Computer Software and Commercial Computer Software Documentation are being licensed to U.S. Government end users (A) only as Commercial Items and (B) with only those rights as are granted to all other end users pursuant to the terms and conditions herein. Unpublished-rights reserved under the copyright laws of the United States. AgentSheets, Inc., 6560 Gunpark Dr., Suite D, Boulder, CO 80301.

10. **COMPLIANCE WITH LICENSES**. If you are a business or organization, you agree that upon request of AgentSheets or AgentSheets' authorized representative, you will within thirty (30) days fully document and certify that your use of any and all AgentSheets software at the time of the request is in conformity with your valid licenses from AgentSheets.

Sim Whatever.

AgentSheets®

Corporate Headquarters
AgentSheets Inc.
6560 Gunpark Dr. Suite D
Boulder, CO 80301
USA
http://www.agentsheets.com
Tel: (303) 530-1773
Fax: (303) 530-3018

European Demo Center
Fraunhofer-Institute for Computer Graphics
Rundeturmstraße 6
D- 64283 Darmstadt
Germany
http://www.igd.fhg.de
Tel: + 49 6151 155 0
Fax + 49 6151 155 199