Wide Ruled: An Author Goal-Based Interactive Story Generator

James Skorupski

expressive intelligence studio

UC Santa Cruz
February 15, 2011
jskorups@soe.ucsc.edu

http://eis.ucsc.edu/Wide_Ruled
Wide Ruled

- Text-based interactive story generator with GUI
- Modified *UNIVERSE*-style story model (Michael Lebowitz)
  - Hierarchical, author goal based story structure
  - Originally designed for soap opera style serial storylines (never ending!)

[Diagram of a text-based interactive story generator with GUI]

[Table of traits for John Smith]

[Table of relationships for John Smith]
UNIVERSE Overview

- Universe an **author modeling** story system
- Author plans and plot fragments that can coordinate one, many, or no characters
  - Author goals and plans may make no sense from character viewpoint
- Associated with each plot fragment
  - Author goal it can achieve
  - Characters
  - Constraints
  - Ordered list of subgoals (steps)
Example character in Universe

Name: Liz Chandler

Marriages:
  - Don Craig [1980]
  - Tony Dimera

Stereotypes: Actor, Knockout, Socialite, Party-goer

Trait modifiers: (Sex F) (Age young-adult) (Wealth 3) (Promiscuity -3) (Intelligence 3)

Description:

- Wealth 8
- Promiscuity 3
- Competence NIL
- Niceness 0
- Self-Conf 6
- Guile 7
- Naiveté 7
- Moodiness 6
- Phys-Att 7
- Intelligence 7
- Age young-adult
- Sex F

Goals: (Find-Happiness Become-Famous Meet-Famous-People)
Example plot fragment

Plot fragment: forced-marriage
Constraints: (has-husband ?her) (has-parent ?husband)
            ( < (trait-value ?parent 'niceness) -5)
            (female-adult ?her) (male-adult ?him)
Goals: (churn ?him ?her) {prevent them from being happy}
Subgoals: (do-threaten ?parent ?her “forget it”)
          (dump-lover ?her ?him)
          (worry-about ?him)
          (together * ?him)
          (eliminate ?parent)
          (do-divorce ?husband ?her)
          (or (churn ?him ?her)
               (together ?her ?him))
No more lisp

- Wide Ruled is a “structured editor” - you have access to a UNIVERSE-like story model, but you can’t write “wrong” code

Plot fragment: forced-marriage
Constraints: (has-husband ?her) (has-parent ?husband)
(trait-value ?parent ‘niceness) -5)
(female-adult ?her) (male-adult ?him)
Goals: (churn ?him ?her) {prevent them from being happy}
Story Components

Wide Ruled stories are made of:

- **Objects** ("What" of the story)
  - **Characters, Environments** – attributes and relationships
  - **Plot Points** – major episodic story information

- **Structure that acts on these objects** ("How" of the story)
  - **Author Goals** – What to do in the story
  - **Plot Fragments** – Ways of achieving the goals
What

Characters
- John Smith
- Jerry Fontana
- Cindy Rollins
- Gene Franks
- Rachel Delores
- Dick Tracy
- Sherlock Holmes

Environments
- The Docks
- The Alleyway
- The City Park
- The Shady Bar
- The Abandoned Shack

Plot Point Types
- Crime Info
- Murderer
- Murder Solved

Goals and Plot Fragments
- Do Murder Mystery
- Do Story
- No more alive victims
- Do the Crime (Text Victim)
- Random Murderer
- Murderer is Enemy
- Attack the Victim (Text Victim)
- Gunshot
- Stabbing
- Find a Detective (Text Detective)
- Detective sleeping
- Detective drunk
- Investigate
- Find crime scene evidence
- Interview friend
- Interview coworker
- Find clue at murderer
- Put together clues and
- Solve the Crime
- Solve with celebration
- Solve with smugness
- Capture the Murderer
How
Characters and Environments

- Object with two lists:
  - Traits (number, text, true/false)
  - Relationships (with strengths)

- Relationships connect objects of same type:
  - Character-to-character
  - Environment-to-environment
Plot Points

- Plot Points are objects containing important story information:
  - Murder mystery: the murderer, the victim, the detective

- Plot Points are temporary, and can only be created, edited, or deleted during story generation

- You specify what they can contain before story generation

- ...but you fill them with information within Plot Fragments
Author Goals
- Reflect story structure, high level intent of author
- Can involve multiple characters, environments (or none)
- **Murder mystery example:**
  “Murder Victim”, “Investigate”, “Capture criminal”

Plot Fragments
- Different ways to achieve a **single** Author Goal
  - A precondition
  - List of story actions to be completed in order
  - Can activate another author goal within it (**subgoal**)
- **Example fragments for “Investigate”:**
  “Interview family”, “Run DNA tests”, “Gather suspects”
Murder Mystery Story Sample Goal

Do the Crime

Random Murderer
- Attack the Victim
  - Gunshot
  - Stabbing

Murderer is Enemy
- Attack the Victim
  - Gunshot
  - Stabbing
Murder Mystery Goal/Fragment Hierarchy

- **Do the Crime**
  - Random Murderer
    - Attack the Victim
      - Gunshot
      - Stabbing
    - Murderer is Enemy
      - Attack the Victim
      - Gunshot
      - Stabbing
  - Find a Detective
    - Find crime scene evidence
    - Interview friend
    - Investigate
  - Investigate
    - Investigate
    - Find clue at Murderer's hideout
    - Put together clues and finish investigation
  - Solve the Crime
    - Solve with celebration
    - Solve with smugness
  - Capture the Murderer
    - Capture peacefully
    - Capture with a fight
  - Restart Story
    - Choose another victim and restart

- **Do Story**
  - No More Victims, end of story

- **Do Murder Mystery**
How do we make a single story?

1. Start with initial Author Goal, selected by author
2. Generator looks at every Plot Fragment for that Author goal
3. Generator checks preconditions for Plot Fragments
4. Generator picks **one** Plot Fragment with satisfied preconditions, and then executes every story action **in order**
   - If a story action is a **subgoal action** go to step 2

A plot fragment can only be selected if it’s entire precondition is satisfied
Author Goals and Plot Fragments - Review

- **Author Goals**
  - Reflect story structure, high level intent of author

- **Plot Fragments**
  - Are different ways to achieve a single Author Goal
  - Contain:
    - Precondition
    - Ordered list of story actions
      - print text, calculate, edit character/environment/plot point, create/delete plot point, and **subgoal**
Author Goal Editor

- Author goals can have parameters that pass information to plot fragments
Plot Fragments

- The Plot Fragment editor

![Diagram of the Plot Fragment editor](image)
Preconditions

- A Plot Fragment is chosen if its **Precondition** is satisfied
- Each constraint statements matches against
  - Characters
  - Environments
  - Plot Points
- Precondition is a list of constraint statements that must **ALL** be true (each statement has at least one char./environ./plot point that is compatible with the constraints)

There exists a Character, where Trait "Name" == Victim Name, and Relationship "Enemy" target name is saved as variable "enemyName" and Relationship "Enemy" strength > 5
Preconditions and Story Actions

- **Preconditions capture** a subset of characters, environments, and plot points with specific attributes and relationships (character with a certain age, or environment with a certain weather, etc)
**Preconditions and Story Actions**

- *Preconditions capture* a subset of characters, environments, and plot points with specific attributes and relationships (character with a certain age, or environment with a certain weather, etc)
Preconditions and Story Actions

- **Preconditions capture** a subset of characters, environments, and plot points with specific attributes and relationships (character with a certain age, or environment with a certain weather, etc)
Preconditions and Story Actions

- **Preconditions** capture a subset of characters, environments, and plot points with specific attributes and relationships (character with a certain age, or environment with a certain weather, etc)
There exists a Character, where Trait "Name" == Victim Name, and Relationship "Enemy" target name is saved as variable "enemyName" and Relationship "Enemy" strength > 5
Preconditions and Story Actions

- **Story actions** do things with those things, for example:
  - Create a plot point and put things into it
  - Change the age/name/relationships of a character
  - Calculate a new number variable: \( \text{varY} = \text{varX} + 1 \)
  - Print out text (how the story is made!)
  - Activate another author goal

Blah, blah, blah, blah ...
Story Actions

- Ordered list

Display Text: "The time: <crimeTime>, the place: <crimeSceneDescription>.

Pursue Subgoal "Attack the Victim (Victim Name)"

Display Text: "<enemyName> appears out of the darkness as the evil murderer, and escapes into the distance inevitable eyes of the police."

Create new Crime Info Plot Point, saved as variable "crimeInformation"

Edit saved Crime Info Plot Point "crimeInformation": set trait "Victim Name" to saved variable "Victim Name"

Edit saved Crime Info Plot Point "crimeInformation": set trait "Crime Scene Name" to saved variable "crimeSceneName"

Create new Murderer Plot Point, saved as variable "murdererInfo"

Edit saved Murderer Plot Point "murdererInfo": set trait "Murderer Name" to saved variable "enemyName"

Edit saved Murderer Plot Point "murdererInfo": set trait "Murder Hideout Name" to saved variable "hideoutName"

Display Text: "<detectiveName> decides to interview <friendName>, a friend of <victimName> murdered. He finds that <victimName> and <murdererName> had a recent argument over money"

Calculate: newClueCount = currentClueCount + 1

Edit saved Murderer Plot Point "murdererInfo": set trait "Clues Found" to saved variable "newClueCount"

Pursue Subgoal "Investigate"
Story Actions

- Display Text
- Pursue Subgoal (any Author Goal)
- Calculate new value
- Create Character/Environment/Plot point
- Edit Character/Environment/Plot point
- Delete Character/Environment/Plot point
Variable Binding

- Variables are pieces of saved and named data that can be used in various places in a plot fragment.
- They come from goal parameters, the precondition, and some actions (calculation, create new plot point)

Variables can be used from Left to Right, top to Bottom (they must be created before using them)
Variable Binding – Precondition

- Precondition Statements can save traits, relationship target names, and relationship strength values to variables for later use
  - Save trait: There exists an Environment, where Trait “Description” is saved as variable “crimeSceneDescription”
  - Save entire object: There exists a Character, saved as “victim”.

- Create a variable and then use it again for comparison
  - Example: Match two characters with different names, and one is younger than the other. Two precondition statements, in order:
    - 1. There exists a Character, where Trait name is saved as variable “nameVar”, and Trait Age is saved as variable ‘friendAge’
    - 2. There exists a Character, where Trait name != nameVar, and Trait Age < friendAge
Variable Binding – Precondition

Edit Environment Precondition Statement
This Environment must exist

Constraints
- Trait "Potential Murder Location" == True
- Trait "Name" ==
- Trait "Description" is saved as variable 'crimeSceneDescription'
- Trait "Time" is saved as variable "crimeTime"
- Relationship "Nearby Location" target name is saved as variable

Constraint Details:
- Type of attribute to match: Trait
- Description: Save Only
- Value saved as: crimeSceneDescription
- Use Saved Variable

Save Environment ...

Edit this Variable
- Must always be true

Save this Value ...
- Must always be true

expressiveintelligencestudio

UC Santa Cruz
Variable Binding

Gunshot

Author Goal
Attack the Victim

Parameters
Name | Type
---|---
victim name | Text

Precondition
There exists a Character, saved as "victim", where The

Story Actions
Display Text: "<victim name> turns around suddenly and hangs erupts and our victim falls flat to the ground, cc
Edit saved character "victim": set trait "Alive" to False

expressiveintelligencesstudio
UC Santa Cruz
Making the story appear

- **Displaying Text** - printed text that can contain captured information (variables) from the rest of the plot fragment

<victim name> turns around suddenly at the sound of the click. Just as <victim name> notices the gleam of a barrel, a loud bang erupts and our victim falls flat to the ground, coughing and sputtering up the last bit of an ending life.
Making the story appear

- The time: 4 am, the place: the large, open city park in the middle of the city. Rachel Delores turns around suddenly at the sound of the click. Just as Rachel notices the gleam of a barrel, a loud bang erupts and our victim falls flat to the ground, coughing and sputtering up the last bit of an ending life. Jerry Fontana appears out of the darkness as the evil murderer, and escapes into the distance to seek shelter from the inevitable eyes of the police. Cut to a quiet single bedroom apartment. The shrill ring of a phone awakens Sherlock Holmes from a deep and relentless slumber ...
The time: 9 pm, the place: a dark rain-drenched alley off 15th street. Kim Bates turns around suddenly at the sound of the click. Just as Kim notices the gleam of a barrel, a loud bang erupts and our victim falls flat to the ground, coughing and sputtering up the last bit of an ending life. Fred Gordon appears out of the darkness as the evil murderer, and escapes into the distance to seek shelter from the inevitable eyes of the police. Cut to a quiet single bedroom apartment. The shrill ring of a phone awakens Dick Tracy from a deep and relentless slumber ...
Plot Fragments

Random Murderer

Author Goal
Do the Crime
Parameters

Precondition

Story Actions

Display Text: "The time: `<crimeTime>`, the place: `<crimeLocation>`
Pursue Subgoal "Attack the Victim (Victim Name)"
Display Text: "<victimName> appears out of the dark, inevitable eyes of the police."
Create new Crime Info Plot Point, saved as variable "crimeInfo"
Edit saved Crime Info Plot Point "crimeInformation": <crimeInfo>
Edit saved Crime Info Plot Point "crimeInformation": <crimeInfo>
Create new Murderer Plot Point, saved as variable "murderer"
Edit saved Murderer Plot Point "murderInformation": <murderer>
Edit saved Murderer Plot Point "murderInformation": <murderer>

Name | Type
--- | ---

Change Goal ...
New ...
Edit ...
Copy
Delete

OK
A single story is generated on the fly, to a text box, every time you click “Generate Story”.

It is slowed down, so readers can interact with it (next slide).
Interactivity

- Interactive Actions
  - Third person, Asynchronous
  - Author-specified **goals** which reader can activate **at any time**
  - Reader can **interrupt** the story at any time and perform meaningful, author-designed actions
  - Murder mystery examples: “Kill the detective”, “Change the murderer”, etc
Interactivity

- What are “Interactive Actions”?

- Just regular author goals with plot fragments, that the author allows to be activated during the reading of the story.

- These goals are executed completely before continuing with the previous story:
  - Any changes to the story world are kept.
  - All subgoals are activated (this could create very long “interruptions” of the story).
......He accepts his fate quickly, picks up the phone, and gathers the details about his new case. Once the call is finished, he jumps in the car, and heads to the crime scene. The facts: the victim is Rachel Delores, a 51-year old woman from Lyndhurst, New Jersey, found dead earlier today.

Cut to the dark hideout of our ruthless killer. Before sleeping for the night, Jerry Fontana stares coldly into a bathroom mirror and removes a thin latex mask. Underneath the disguise appears the face of John Smith, the true villain in this story.

Sherlock Holmes decides to interview John Smith, a friend of Rachel, for any clues as to why our victim was murdered. He finds that Rachel and John had a recent argument over money ...
Misc Plot Fragment Tasks

- Must selecting **start goal**: use “Start Goal” check box, can’t have any parameters (by default, start goal is first one you create)

- **Creating** char’s/env’s/plot points within a Plot Fragment:
  - 1. **New Action** “Create New …”, save as a variable name, fill it with static numbers or strings
  - 2. **Edit Action**: To insert variables, perform edit action on char/env/pp saved in creation action,

- **Deleting** char’s/env/plot points within Plot Fragment
  - 1. Match in precondition, save to variable
  - 2. Use **Delete Action** on variable

- **Editing** char’s/env’s/plot points
  - 1. Match in precondition, save to variable
  - 2. For **every** trait/relationship to edit, you create one **Edit Action**, and select that saved variable in step 1.
Notes

- Website: [http://eis.ucsc.edu/Wide_Ruled](http://eis.ucsc.edu/Wide_Ruled) (Click on Project Page link for download/installation info)
- Tutorial posted online – Step by step instructions
- Documentation, sample story online, with these slides
- Wide Ruled is Windows XP, Vista, 7 only
  - Installation instructions posted online
  - Uses .NET 3.5 – need version 3.5 or later (4.0)
  - Works with Parallels, VMWare Fusion on OS X, VirtualBox (probably)
The future of Wide Ruled: **Story Canvas**
Story Canvas

- Visual, Storyboard-based authoring
- Same UNIVERSE-like story model, new graphical interface
Story Canvas

- Editing Plot Fragments

Precondition

Actions

Goal 1
- Fragment 1
- Fragment 2
- Fragment 3

Goal 2
- Fragment 4
- Fragment 5

The time: 8 pm, the place: a secluded beach at sunset. <name> and <name> meet each other...

Subgoal: Select Annoying Friend

Hey <name>!

Who's that creep?

Later that night...

Where have you been?

Subgoal: Introduce Conflict
Story Canvas

- Reading generated stories

The time: 8 pm, the place: a secluded beach at sunset. Julie and Mark meet each other...

Nice sunset eh?
My name is Mark, yours?

I think I'll go to the beach.

Ugh!
Who's that creep?

Hey Julie!

Later that night...
Where have you been?!
Initial Evaluations this quarter (stability and basic functionality testing)

Full evaluations in Spring – we need you!

- Interested students can make their Wide Ruled stories in Story Canvas
- Help us research!
- If you are interested, we will send out emails in a few weeks
Wide Ruled Deliverable

- .wr2 file emailed to Ben
- Requirements
  - Must run (must generate story)
  - Must use every feature at least once
    - Characters – traits and relationships
    - Environments – traits and relationships
    - Plot Points – traits
    - Author goal – parameters
    - Plot fragments
      - Preconditions
        - Character precondition, environment prec, plot point prec.
        - Save a trait to a variable
        - Use variable within precondition
      - Actions
        - Print text (with inserted variables), subgoal, calculate value, edit character, edit environment, create plot point, edit plot point, delete plot point
      - Interactive Actions – at least one action that changes the story outcome
Contact

- Email support: jskorups@soe.ucsc.edu