Entity-RFC

Metacosm Development Team

Revision History

Revision M1 July 15th, 2001
Added material on Entity creation and Influences, editor use cases, docbook conversion, misc corrections and synchronization with other RFCs
Revision 0.5 November, 28th 2000 Revised by: Entropizer
Small corrections and clarifications
Revision 0.4 November, 6th 2000 Revised by: Entropizer
Rework of the whole document: removed Characterization Levels and Characterization Pools, added Influences and Types, reworked Definitions, several fixes
Revision 0.3 February, 29th 2000 Revised by: Entropizer
More examples, included elements from discussion on the mailing list, changed Strength into Power
Revision 0.2 February, 21th 2000 Revised by: Entropizer
"Presidents' Day release" First SGML version, corrections and expansions
Revision 0.1 January, 2000 Revised by: Entropizer
First version

This document is meant to gather information on what an entity will be. It is based on a former RFC named Population RFC. It casts a new look on this former proposal and expands several aspects of it so as to create a consistent and dedicated RFC that solely deals with what is an Entity.

1. License

2. Metacosm

For more information on Metacosm, check our site: Metacosm (http://metacosm.org).

3. Definitions

3.1. Entity

Entity is the basic concept of Metacosm game engine. It is therefore extremely important to clearly identify what makes an Entity.

Entity

An Entity is an element of the framework that can be manipulated or interacted with during a game session.

Entities

Entities = Places + Creatures + Items

See Also: PlaceCreatureItem.

Therefore, since Metacosm goals include a high-playability and interactivity, almost anything in the game (Items, Creatures, Places...) is to be implemented as an Entity.

3.2. Properties

Cf Property-RFC

3.3. Characteristics

Characteristic

Characteristics describe an Entity from an objective point of view. They gather parameters that can be manipulated via a set of rules. Characteristics represent the core of the information manipulated by the framework when a world is simulated in Metacosm. Characteristic is a generic term for Description, Identity, Attributes, Capabilities and Skills.

For more details, see the Characteristics section.

3.4. Attributes

Attribute

Attributes define the inherent qualities of an entity. They describe Entities using Properties thus providing an objective description of an Entity. Attributes are basic descriptive elements of Entities. This description is not targeted at Players but at the simulation engine.

For more details, see the Attributes section.

3.5. Capabilities

Capability

> Capabilities are derived Attributes. Capabilities are Properties that are calculated instead of being fixed and then modified. They are based on combinations of Attributes. They reflect an Entity's natural ability to perform certain types of actions, regardless of skills.

Implementation: a list of couples (coefficient, Property/Capability name)

For more details, see the Capabilities section.

3.6. Community

Community

> Community is an abstract concept introduced to talk about a group of Entities. The word "community" is to be interpreted in a very general, abstract way since it doesn't imply anything else on member Entities than commonality between them.

An Entity is part of a given Community if it is influenced by the Influence defining the Community i.e. the Community exists only as long as several Entities are influenced by the same Influence. If an Entity isn't influenced anymore by a given Influence, it leaves the Community. Communities are implicit and only defined by the Influence. An Entity can be part of several different Communities.

3.7. Influence

Influence

> An Influence defines a Community of similar Entities and how being part this Community "influences" Entities that belongs to it. An Influence gathers Characteristics and other game constructs that are common to the Community of Entities it defines, thus modifying Entities it influences. We use the term "influenced" when referring to game constructs that are modified ("influenced") by Influences and "influenceable" for game constructs that can be "influenced".

For more details, see the Influences section.

3.8. Type

Type

> A Type is a special kind of Influence. It is an Influence in the sense that it gathers Characteristics and modify Entities. It differs from simple Influences by its focus. While simple Influences modify already created Entities, the purpose of Type is to create new Entities.

4. Influences

Communities share common knowledge and/or common characteristics. As far as Metacosm is concerned, this can be modelled by an Influence that gathers the common knowledge and expresses the common characteristics in terms of game constructs (Characteristics, Actions, descriptions, etc). Characteristics define the essence of the Entity. These can be very different from an Entity to another in their number, type or value. What makes an Entity different from another is the set of Influences that "influence" its Characteristics. The way that Entities are influenced greatly vary from an Entity to another depending on the number or type of Influences that act on it.

As defined above, Influences define a Community of Entities that are influenced in a similar way. An Influence gathers everything that makes the Entities that it influences special and distinguish them from members of other influenced Communities. It is worth noting that an Entity can belong to several influenced Communities (for cumulative effect) and that influenced Communities can overlap or coincide.

An Influence can modify an Entity in the following ways:

- A numerical or text Property may be set to the value of an expression containing constant values, other Properties, standard operators (+, -, *, /, % modulo, statistical functions, string equality, search, replace...). Typical use include absolute modifier (new value does not depends on current value) and relative modifier (new value is a function of the current value).

  If the expression result is outside the field of acceptable values, the *closest* acceptable values is used.

- The field of numerical Properties may be scaled by a real factor or translated by a real value.
- A new Property may be defined. Initial field and value could be provided. Note that they can be derived from existing Properties through the use of expression.
- The value of an existing Property can be changed: the description of the change is stored in a structure called a PropertyModifier. If the new value depends on the current one, the modifier is said to be Relative. If the new value is independent of the current one, the modifier is said to be Absolute.

  At any time, the effective value of a Property is defined by the Absolute Modifier of greatest priority and all Relative Modifier which priority are larger than the Absolute Modifier Priority. Relative modifier are applied ordered be priorities, lower priorities first.

- New Actions can be added to the list of ones potentially performed by the entity.

- The handling of Actions can be modified. Cf Interactions-RFC.
- An Access to a shared ’community’ memory can be provided.
- The description of the Entity can be changed or completed.

Influence Actions on Entity obeys to a rule of priority: the list of Influences acting on an Entity can be seen as a set, the Influences with greater priority apply.

For the sake of understanding, Influence are classified by "theme": for example, Type/Species, Geography, Activity/Job, Individual experience. This classification will become a whole (forest (i.e. several trees) like) hierarchy later when it’s needed.

At Entity creation time, some parameters will be choosen randomly and stored in the Entity instance. All other data used to compute Properties values will be extracted from Influences instances.

An Influence may only be applied on an Entity respecting a set of constraints (set of preexisting Influence or Properties, some Properties values must belong to some defined field...). See Predicate-RFC, to see how these constraints are expressed.

Example 1. Midgaard inhabitants

The inhabitants of Midgaard know the city better than the average tourist and can therefore more easily find their way. They might also know that Raoul is the baker that lives in the northern part of Midgaard. They might be particularly resistant to the locally brewed beer making it difficult for wannabe adventurers to get them drunk in order to get information... and so on.

Example 2. King Solomon’s palace

King Solomon’s palace rooms all share a particular architecture and design. They are all influenced by the same Influence, they are all part of the same Community. Moreover, among these rooms, King Solomon has had some of them decorated in yellow using only a particular kind of fabric so as to create a "Yellow Suite" Community inside the King Solomon’s palace rooms Community, each of these Communities being influenced differently.

We think that this concept brings flexibility in Entity management.

We next distinguish between required and optional Influences. Required Influences are Influences without which an Entity cannot exist. They are at the base of all Entities. There are currently two required Influences. Optional Influences are limited by game designers' imagination only.

4.1. Required Influences

4.1.1. Common Influence

The Common Influence defines what is is shared by *all* Entities. This is the minimum required from a game object to be recognized as an Entity. The Common Influence defines a set of Attributes that are common to all Entities or to put it differently, all Entities are part of the Common Community. At the implementation level, it also defines a standard interface to interact with an Entity regardless of its implementation.

There should be exactly one Common Influence per instance. That Common Influence will specify the attributes of all Entities in the world of this instance and will be world- and rule-specific. More precisely, if you want to create a Shadowrun-based world, the Common Influence will describe everything that is common to all players in a Shadowrun game (all the attributes). If you play in DnD-based world, the Common Influence will be totally different. [Shadowrun is a Fasa Corporation trademark and DnD is a TSR trademark]. What this means is that the Common Influence kinda describes a character sheet for a traditional RPG. Each RPG has its specific character sheet (referred to as CS thereafter) which is then filled up according to this RPG's rules. Now, how these CS are filled up is depending on the type of character in traditional RPG. It's also true for Items or Places.

How the CS is filled in Metacosm is determined by the Entity Type. The Type is also an Entity. More precisely, the Type IS the typical Entity of this Type, it perfectly describes the average Entity of this Type. (i.e. the Type is an Entity which Type is itself). To create a new Entity of this Type, the Type is cloned and then its attributes and parameters are randomized according to this Type distribution laws (or otherwise:

simulation of number of dices, fixed, etc) and influenced by the creation location (Place) of the Entity. This makes for a pretty simple and powerful system.

An Entity is affected once and only once by a Common Influence.

4.1.2. Type

Types are a special kind of Influence. Entities are created by Types which initialize the Characteristics defined by the Common Influence and add new ones when needed. A Type can reference several different other Influences that are "applied" to an Entity at creation time.

An Entity is affected once and only once by a Type Influence.

4.2. Optional Influences

Optional Influences are gathered by levels. Each level of Influence represents a particular kind of Influence. We present a list of possible Influence levels here. This list is not necessarily exhaustive.

An Entity can only be affected once by a given Influence. It can be affected by as many as it wants/needs different Influences.

4.2.1. Geography

This level gather Influences that modify all Entities found in a given area. There should be rules to establish how geography can influence Entities. We can identify two different modalities for Geography Influences: static or dynamic.

The static modality determines the Entity's parameters at creation time. Static Influence will be handled by Types since Types have the responsibility for creating Entities.

Geography can also influence the development of an Entity after its creation. When an Entity able to evolve is put in a given environment, it will slowly adapt itself to the conditions of that environment so as to optimize its probability of survival.

Every Entity can potentially be influenced by its location at creation time (static Influence). However, only those Entities that can evolve can be dynamically influenced by their environment.

Example 3. Static influence

The composition of the area in which a rock is created, influences the very essence of this rock. Similarly, someone born in a rather cold environment would be more prone to better resist cold...

Example 4. Dynamic influence:

Once upon a time, short-haired rabbits were dominant. But now, these days are over since the planet's inclination has changed slightly, hardening the climate and bringing in cold and ice. Rabbits with longer hair are now more able to resist the low temperatures and the rabbit population slowly evolves to grow longer hair...

4.2.2. Individual

As for geography, an Entity can be affected both statically and dynamically at the individual level. When an Entity is created (static modality), it is created with "unique" Characteristics. Exactly how unique those Characteristics are is irrelevant here. It could however happen that two distinct Entity (with different identities, see "Characteristics" for a discussion of identity) have the same Characteristics (twins). If these two are unable to evolve, they will forever be exactly similar. However, if they are able to evolve, they will go through their life experiencing different things and will be dynamically influenced by their deeds and/or their environment at the Individual level. Even though most of the way how an Entity will evolve will be defined by its Type or at the Community level, individual variations can occur. This is of crucial importance for Characters.

Example 5. Non exhaustive list

Diseases, injures, maledictions, benedictions, spells, drugs, magic items, inventory/possessions (weight, volume of possessions, blindfold, etc)

4.3. Examples

Note that this examples are not necessarily representative of an actual implementation.

Example 6. Elektra's wand

Glass is a fragile matter so a wand made out of glass is rather fragile (-3 in endurance, Type level). However, Elektra, Mistress of illusions, had her glass wand created with a specific process and worked on the glass composition to make it more resistant (+2 in endurance, Individual level).

- Type level: made of glass (endurance -3)
- Individual level: special creation process (endurance +2)
- Result: endurance -1 (-3+2)

Example 7. Earendil the Elf

Earendil is an elf so he is less strong than an average human by heredity (-1 in power, Type level). However, he is a member of a rare elf cast who are stronger than average elves (+1 in power, Community level).

- Type level: elf (power -1)
- Community level: member of a stronger elfin cast (power +1)
- Result: power +0 (-1+1)

Example 8. Adamant

Adamant is Hercules' son. He has thus inherited the physical power of his father (power +5, Individual level). However, he has been cursed by Hera and he's bound to live under the sea which reduces his ability to affect other entities (power -3, Geography level). Moreover, he is suffering from a cold decreasing momentarily his power (power -1, Individual level).

- Individual level: son of Hercules (power +5)
- Geographical level: lives undersea (power -3)
- Individual level: sick (power -1)
- Result: power +1 (+5-3-1)

5. Characteristics

5.1. Identity

One of the framework goals is improved realism. To ensure world coherence, each creature needs to be identified so that each of its modification could be made permanent. For instance, if a Player gives an Item to a Creature, this one will still be in possession of this Item at their next meeting.

Identity cannot be "influenced". We distinguish two different aspects of Identity: internal and external. Internal Identity will be used by game mechanisms to allow world coherence, correct persistence of Entities, efficient identification of an Entity, etc. External identity will be manipulated by Entities themselves. External identity includes name, title, nicknames, etc.

5.2. Description

Cf Perceptions-RFC

5.3. Attributes

Attributes provide the basic objective description of an Entity. Attributes are the target for Influences actions. We will focus here almost exclusively on what is common to all Entities and let the details of specifying other Attributes to other, more appropriate documents.

5.3.1. Common Attributes

The Common Influence defines the set of Attributes that all Entities, regardless of their Type, share with others. Most of the Interactions between Entities should be resolved at the Common level for implementation purposes. Game designers need to be careful in their identification and description. Attributes should make sense regardless of the Entity Type. However, if an Attribute doesn't make sense for an Entity, then this Attribute will just be null for this particular kind of Entities.

Agreeing on the set of Attributes provided by the Common influence is really important and game designers should try to make sure that it will not change since most of interactions will rely on those sole Attributes. Some interactions could call more specific Attributes but they should be careful in that matter and try to rely only on the most common basis. Next, we propose a set of possible Attributes.

5.3.1.1. Endurance

Endurance

> Endurance represents an Entity's resistance to outside forces. The greater the endurance, the better an Entity will resist to anything that can affect its physical state. For a living creature, changes in physical state involve injuries, fatigue or

> illness. For an inert Entity, its endurance represents its solidity (an Item with a high endurance will be harder to break), its resistance to external elements (rain, acid, fire...).
>
> From Merriam-Webster dictionary: Ability to withstand hardship or adversity; especially: the ability to sustain a prolonged stressful effort or activity.

Example 9. Raoul the baker

Raoul the baker is no athlete but he has to work hard everyday in the bakery, moving flour bags, working on the dough in hot environment so he has a slightly higher endurance an average human's.

Example 10. A cave

A cave in a granite cliff has a high endurance since it is very difficult to modify its size (it involves a lot of effort to cave it even more...).

An interesting idea could be to define a more complex endurance that will define distinct endurance for different elements (fire, water, earth, air).

5.3.1.2. Power

Power

> Power represents an Entity's power to physically interact with other Entities. Basically, power represents what is physically acting in the essence of an Entity. It is against what endurance of an Entity has to resist when two Entities are in opposing physical interaction. For a living creature, power represents its physical power, its strength, just how strong it is. For an inert Entity, power represent how potent it is in modifying other Entities physical structure.

From Merriam-Webster dictionary: Ability to act or produce an effect.

Example 11. Behemoth

Behemoth is an impressive Minotaur with muscles therefore, Behemoth will have a high power.

Example 12. A sharp sword

A sharp sword will have a higher power attribute than a staff since it has a higher ability to damage other Entities. Similarly, an acid can have a power just as can a perfume.

5.3.1.3. Coordination

Coordination

Coordination represents an Entity's capacity to act efficiently and harmoniously. For a living creature, coordination represents its agility, its suppleness. It represents just how efficiently it is able to move or perform complex moves. For an inert Entity, coordination represents how well it is built or how well its part fits together. Obviously, a one-part Item will have a null coordination. Only Entities with a specific coordination level would be able to react appropriately to orders given by Controllers. Coordination will be used when Item are animated by spells and affect what the Item can do once animated.

From Merriam-Webster dictionary: Harmonious functioning of parts for effective results.

Example 13. Shankra the Yogi

Shankra the Yogi has spent years and years to gain perfect control of his body. He is therefore likely to have a high coordination. On the contrary, Bert (a well known couch potato) hasn't exercized much since he bought a TV set, so odds are that he won't have a good coordination attribute.

Example 14. An armor

The coordination of an armor will reflect how well it fits Entities that wear it. The coordination of a car is how smooth it will drive and handle the road.

5.3.1.4. Charisma

Charisma

> Charisma represents how an Entity appears to others. the charisma of an Entity is not merely based on its appearance but reflects the capacity that an Entity has to attract positive reactions. For a living creature, charisma represents all that makes the Entity appealing: its beauty, its charm, its animal magic. The same applies also for inert Entities for which charisma can be considered as the way they can appeal to other Entities even when they aren't good looking. The charisma attribute of an Item can be thought as its potential artistic value and as something that will influence its commercial value. Characters will feel more comfortable with Entities with high charisma whereas they could feel that something is wrong with an Entity that has a low charisma. Spells can influence the charisma of Item (curse, "beautifiers"...).
>
> From Merriam-Webster dictionary: A special magnetic charm or appeal.

Example 15. Raoul the baker

Raoul the baker is a rather solitary person that is more comfortable dealing with flour and making breads than conversing with people that find him a little strange, he obviously doesn't have a good charisma.

Example 16. A special necklace

The necklace in Philip K. Dick's novel The Man in the High Castle has obviously an intrinsic charm that is not only quantified by its appearance: it appeals to people, they feel comfortable when holding it, it has a high charisma. A well decorated room with good selected furniture and lights makes people feel comfortable when they stay in that room that seems to have a good charisma. An Item that has obviously a very high charisma is the Unique Ring created by Sauron in Tolkien's Lord of the Rings.

5.3.1.5. Intelligence

Intelligence

> Intelligence represents an Entity's ability to take into account its surroundings and react to it in a sensible way, capacity to solve problems. The intelligence concept is pretty obvious for living creatures. It is a little bit more complex for inert Entities. Obviously, common Items will have a null intelligence. However, technology or magic can make Items more able to handle their surrounding effectively given them a kind of intelligence.
>
> From Merriam-Webster dictionary: Ability to learn or understand or to deal with new or trying situations.

Example 17. Merlin the Sorcerer

Merlin the great Sorcerer is very smart and that is lucky since he has to handle powerful magic energies. He should have a high intelligence attribute.

Example 18. Java-enhanced appliances

Java-enhanced appliances are smarter than ones without any kind of program.

5.3.1.6. Will

Will

> Will represents an Entity's self-control and power of sticking to its goal. For a living creature, will represents its strength of mind, the way it resists to adverse circumstances, seduction or mental manipulation. It is, once again, more complex for inert Entities. We can consider that an Item will is its inertia, its resistance to resist manipulation. It seems close to endurance but an Item's will is more oriented toward tricks (spells or playing with a mechanism for appropriate Items). Endurance deals more with resistance to destruction.
>
> From Merriam-Webster dictionary: The power of control over one's own actions or emotions.

Example 19. Frodo the Hobbit

Frodo, Bilbo's nephew, has a strong will since he was able to go through all the obstacles he encountered on his quest to destroy the Unique Ring. He was also quite strong in resisting the incredible urge to wearing it.

Example 20. A dwarven lock

A dwarven lock is really hard to pick. We can therefore consider that it has a strong will. Moreover, it has a strong endurance too so that it can resist less subtle attempts to open the door it is guarding.

5.3.1.7. Magic

Magic

> Magic represents an Entity’s "understanding" of magical energies and capacity to control them. Magic is similar for both inert Entities and living creatures. This attribute represents just how good a medium an Entity is for magical energies. It represents the magical intensity that an Entity can put in spells or in resisting magic. An Entity with a null Magic attribute is consider as magically inactive. See the (yet to come) Magic-RFC for more information.
>
> From Merriam-Webster dictionary: The use of means (as charms or spells) believed to have supernatural power over natural forces. (Note that this definition is slightly different from the one given here only as a reference.)

Example 21. Merlin the Sorcerer

Merlin the great Sorcerer has a high control on magical energies. He should therefore have a pretty high Magic attribute.

Example 22. Orichalchum

Orichalchum is well known to have magical properties and is very sought after by mages desiring to put its magical power to work. It is likely that Items made out of Orichalchum will have high magic attributes.

5.3.1.8. Essence

Essence

> Essence represents an Entity’s life energy or inherent magical energy. Essence is what separates inert Entities from living creatures. Inert entities have a null essence attribute whereas it is positive for living creatures. When a living creature loses essence to the point of having a null essence attribute, it becomes an inert Entity with different effects depending on what caused the loss of essence. It can mean death, reification (transformation into an Item as an example due to over-cybernetization), destruction of the Entity, transformation into a living-dead, etc. Essence is the most important attribute for living creatures since processes such as aging, diseases, poisoning, etc. can be modelled via the variation of this attribute. The life cycle can can also be simulated via this attribute by ensuring that constantly decreases, thus forcing living creatures to eat to regenerate their essence by getting a sufficient amount of essence back from the food they eat.
>
> From Merriam-Webster dictionary: The individual, real, or ultimate nature of a thing especially as opposed to its existence

Example 23. Earendil the Elf

Earendil has been injured by Azaroth, his mortal enemy. He is bleeding pretty badly increasing the rate of his natural essence lost. Borg has just gone through surgery to add this last piece of equipment that he needed so bad to make him (it?) a full cyborg. He (it?) is now detached from the need of flesh and his essence attribute is now null.

Food provides an essence amount equals to their essence attribute minus possibly essence lost during food processing.

5.3.2. Type attributes

Each Type defines a law (such as 3D6+2) to initialize each Common Attribute.

5.3.3. Other influenced attributes

Optional Influences cannot introduce new Attributes. They may however apply modifiers and/or constraints to them so that they customize an Entity to reflect how its profession or environment affects it. They will also most likely introduce Capabilities.

5.4. Capabilities

Capabilities are derived Attributes (probably affine functions of Attributes). Since they are based on Attributes, they evolve as these do. They reflect the inherent capabilities of an Entity at each moment (i.e there are calcuted dynamically). An Entity cannot learn a Capability since it is a natural ability. It should be possible to dynamically add Capabilities to an Entity at runtime. Capabilities can be considered as higher-level statistics for an Entity. As other Characteristics, Capabilities can be influenced.

5.4.1. Common capabilities

5.4.2. Type capabilities

Types are the priviledged place to add new Capabilities to Entities. Possibilities for Capabilities are almost endless so we won't try to define all of them. Moreover, Capabilities are highly Type-dependent. We will rather give a couple of examples to demonstrate what a Capability could look like.

Example 24. Capabilities for Characters

- combat: (intelligence + will + coordination) / 2
- reaction: (intelligence + coordination) / 2
- spell-casting: (intelligence + will + magic) / 3

- memory: (intelligence + will)
- running: (endurance + coordination)
- ...

If a Capability is calculated with Properties or Capabilities enclosed in intervals, this Capability is also enclosed in a field (calculated with others intervals). This is just a default field. It’s also possible to fix it to other values.

5.4.3. Other influenced capabilities

Even if most Capabilities will be introduced by Types, it is possible that other Influences will introduce new Capabilities. It is more likely that other Influences will act as modifiers on already defined Capabilities to increase or decrease inherent Capabilities according to the training an Entity had received via its profession or adaptation to its environment. It should therefore be possible for an Influence to re-define the way Capabilities introduced by other Influences are calculated.

6. Making it work

6.1. Creating Entities

Entity creation can happen in two circonstances: during game design within the Entity Editor and during the actual gameplay (character, animal, plant birth, Item production, building construction, etc.).

For the game engine, the entity creation process will always look the same. Obviously at the design stage all parameters will be free of constraints, unlike during gameplay. We will describe examples for Items, Characters and Places in these two cases.

6.1.1. Entity creation in the Editor

The editor will allow to create an Entity from scratch or according to a template. Note that a Template is an editor concept and Type is a game framework concept. The template concept can contain more information than what defines a Type (think clothing, initial inventory, the way the name of the NPC is computed). Here the interest is on the entity framework, so we will do everything from scratch.

Example 25. Some food

We want to create an Item of type "Food". We go to Entity Creation. Select a name "Magic Mushroom". Choose a type in a tree. Select Food in the Item branch. No subtype. Two forms appear: one for the Property defined by "Item", which are exactly the ones of the Common Influence, shared by all Entities. It leaves all of them to default. The second form allows me to choose the value of the energy contained in the Item. It is the single Property defined by the subtype "Food". Type, name and field of the Property are defined by the subtype and I cannot change them at all. I change the default value and give a larger one. I come back to the first form, and choose generate. The form closes and another opens up. The Entity has been created: I see that the entity I just created with the name and the Property value I just created has been given an EntityID.

But the Entity still does not belong to the game because it does not exist anywhere. Then I choose the original location of the entity: I choose a place, and a position inside this place.

Example 26. A weapon

I want to create a weapon. I am back in the Entity Creation dialog. The name I choose for it is "kitchen knife". The Type will be Item, weapon, blade. Weapon and blade are basic Influences, not subtypes: They do not define any new Properties. I choose the values of the standard Item Attributes so that the kitchen knife will be a bad quality weapon. And then I confirm, The Entity is generated and I read its Entity ID. I want to use this Item as decoration in a vampire’s castle. So I change its charisma to a very low level. And I influence it with "Blood". In fact this is a subtype which defines the "bloody" Property. I leave the default value for "bloody". The fact that the knife is bloody and cursed will be automatically added to the blade description.

Example 27. A Character

I want to create a Character I would like to create peasants for my middle age area. I will need quite a number of them, so I will start by defining a template in the editor and then derive(clone) identities from it. I open the template creation dialog. I name it "MyPeasants". The type will be Creature, Character, Civilian, Farmer. Civilian and Farmer do not define any Properties, they exist solely to define the community of civilians and farmers. Then I have to define the default inventory of these Characters, which must contains their clothes, tools, and gold. Money is defined by choosing an amount. All other Items in the inventory are created by cloning existing Items or template.

6.1.2. Entity creation inside the game

All sorts of events can lead to Entity creation in the game:

- Character, god, animal reproduction (individual outside Populations) creates new Characters
- Many actions can change an entity into a new one. For example, death has for consequences the creation of a dead body. The baker uses wood, water, salt and flour to bake bread. A craftsman uses wood and metal to create Items.
- Building cities, roads, and houses leads to the creation of new places

These new entities are influenced by the context in which they were created. For example newborns will be influenced by their genetic heritage (depending on their parents' properties at the time), the place they were born, their zodiacal sign... These influence are not essential and will not define any new Properties by themselves. The action responsible for the creation of the entity will have complete control on how these influences are taken into account. It is possible to get rid of all of them.

Example 28. A birth

Birth of a child in a sorcerer guild. By default the type of the new character is choosen among the type of the parents. The new entity be influenced by:

- the Common influence
- the character influence, and all influence in the branch which leads to its individual type
- an influence corresponding to his/her gender
- an influence corresponding to his family (facultative)
- an influence corresponding to his birth place (facultative)
- an influence corresponding to his zodical sign (..)

7. Entity-Editor

8. Implementation Plans