1. Purpose of Component

This component represents one or more stores of supplementary feed. A component instance represents the stores and paddock-available amounts of several supplements. Each supplement type is distinguished by a name and is represented by the amount in store together with a number of attributes relating to its quality as a diet for animals.

Feed may be bought and then (logically) placed in one of the "paddocks" to which animals in the Stock component may be assigned. Feed which has been placed in a paddock is accessible to grazing stock in that paddock. If more than one supplement is placed into a paddock, the animals access a mixture.

When a "conserve" event is executed (for example from a Pasture component instance), the resulting conserved fodder is transferred to a special "fodder" store in the Supplement component, so making it available for later feeding.

2. Initialisation Properties

Property	Type	Required?	Units	Description
spoilage_time	double	No	d	Time over which an amount of supplement placed in a paddock will become inaccessible to grazing stock. Default value is 0.0, i.e. supplement only persists for the time step that it is fed out.
stores	record[ ]	No		Attributes and initial amount in each supplement store.
: *name*	string			• Name of the store, to be used in *buy* and *feed* events.
: *stored*	double		kg	• Initial amount of supplement in the store (fresh weight basis).
: *roughage*	Boolean			• TRUE i.f.f. the feed is a roughage.
: *dm_content*	double		kg/kg	• Dry matter content of the feed.
: *dmd*	double		-	• Dry matter digestibility of the feed (not including any portion that passes the gut undamaged).
: *me_content*	double		MJ/kg	• Metabolizable energy content of the feed.
: *cp_conc*	double		kg/kg	• Crude protein content of the feed.
: *prot_dg*	double		kg/kg	• Protein degradability of the feed.
: *p_conc*	double		kg/kg	• Phosphorus content of the feed.
: *s_conc*	double		kg/kg	• Sulphur content of the feed.
: *ee_conc*	double		kg/kg	• Ether-extractable content of the feed.
: *adip2cp*	double		kg/kg	• Proportion of crude protein that is insoluble in acid detergent.
: *ash_alk*	double		mol/kg	• Ash alkalinity of the feed.
: *max_passage*	double		kg/kg	• Maximum proportion of the feed that will pass undamaged through the gut of ruminants.

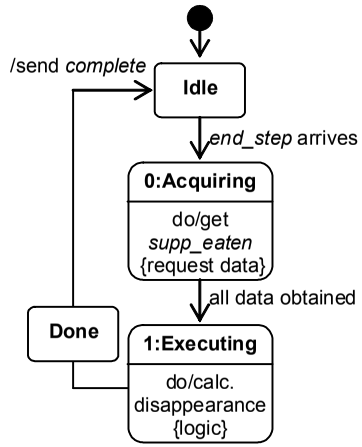
3. Subscribed events – sequenced

3.1. end_step

Default sequencing: 9990

This event determines the amount of supplementary feed eaten by livestock and removes it from the amount present in each paddock. It then computes "spoilage" of supplement.

The event follows the pattern of execution shown at right.



4. Subscribed events - other

4.1. buy

Increases the amount of supplement in a store.

Parameter	Type	Units	Description
supplement	string		Name of the supplement to purchase.
amount	double	kg	Fresh weight of supplement to purchase.

In addition to the names of supplement stores created at initialisation, one of a standard set of supplements may be given in the *name* parameter. In this case, a new supplement store is created and default supplement attributes are used. The standard supplement names are:

'barley, crushed'	'barley, whole'	'barley straw'	'beans, field'	'bran'
'brewers grains'	'canola meal'	'canola straw'	'citrus pulp, dry'	'coconut meal'
'cottonseed, whole'	'cottonseed meal'	'fish meal'	'groundnut meal'	'hay'
'lucerne pellets'	'lupins'	'linseed meal'	'maize'	'maize bran'
'maize gluten fd'	'maize straw'	'molasses'	'oats, crushed'	'oats, whole'
'oat bran'	'oat chaff'	'oat straw'	'peas'	'pea pod meal'
'pea straw'	'pellets'	'rice, crushed'	'rice, whole'	'rice bran'
'rice straw'	'rye'	'safflower meal'	'silage'	'sorghum'
'sorghum, whole'	'soya bean meal'	'sunflower meal'	'tallow'	'triticale, crushed'
'triticale, whole'	'urea'	'vetch grain'	'wheat, crushed'	'wheat, whole'
'wheat chaff'	'wheat pollard'	'wheat straw'		

4.2. feed

Transfers an amount of supplement from store to one of the paddocks, where it will be accessible to grazing stock.

Parameter	Type	Units	Description
supplement	string		Name of the supplement to be fed.
amount	double	kg	Fresh weight of supplement to feed. If the amount nominated exceeds the amount currently in store, the amount in store will be fed.
paddock	double	-	Name of the paddock in which to place the supplement. The paddock name must be either the null string or the name of a component that has the *area* variable.

4.3. mix

Transfers an amount of supplement from one store into another. The transferred supplement is mixed with any supplement already in the destination store.

Parameter	Type	Units	Description
src_store	string		Name of the source supplement store.
amount	double	kg	Fresh weight of supplement to be transferred. If the amount nominated exceeds the amount currently in the source store, the amount in store will be transferred.
dest_store	string	-	Name of the destination store. If the name of the destination store is not recognised, a new store is created.

4.4. on_conserve

Notifies the component that an amount of forage has been conserved.

Parameter	Type	Units	Description
fresh_wt	double	kg	Mass of conserved forage
dm_content	double	kg/kg	Dry matter content
dmd	double	-	Dry matter digestibility
n_conc	double	kg/kg	Nitrogen content
p_conc	double	kg/kg	Phosphorus content
s_conc	double	kg/kg	Sulphur content
ash_alk	double	mol/kg	Ash alkalinity

5. Methods

None.

6. Published events

None.

7. Driving properties

Property	Type	Units	Event:State	Number	Description
supp_eaten	record[ ]		*end_step*:0	0+	Consumption of supplementary feed by animals
: *paddock*	string				• Name of a paddock
: *eaten*	double	kg			• Amount of supplementary feed eaten by animals in this paddock.

In addition, the presence of instances of the *area* variable (double, units ha) is checked during initialisation to locate the paddocks present in the simulation.

8. Owned properties

All initialisation properties are readable. In addition, the following owned properties are available:

(a) Standard properties

Property	Type	Units	Description
name	string		Fully-qualified name of the component.
type	string		Value is "Supplement"
version	string		Value is "1.0"
author	string		Value is "CSIRO Plant Industry"
active	Boolean		Denotes whether or not the component is active
state	string		SDML description of the current state

(b) Component-specific properties

Property	Type	Units	Description
no_stores	integer4		Number of supplement stores.
no_paddocks	integer4		Number of paddocks recognised by the component instance.
padd_names	string[ ]		Name of each paddock recognised by the component instance.
padd_amounts	double[ ]	kg	Amount of supplement currently accessible to stock in each paddock recognised by the component instance.
supp2stock	record[ ]		Amount and attributes of supplementary feed present in each paddock.
: *paddock*	string		• Name of the paddock.
: *amount*	double	kg	• Amount of supplement (fresh weight basis) in the paddock.
: *roughage*	Boolean		• TRUE i.f.f. the feed is a roughage.
: *dm_content*	double	kg/kg	• Dry matter content of the feed.
: *dmd*	double	-	• Dry matter digestibility of the feed (not including any portion that passes the gut undamaged).
: *me_content*	double	MJ/kg	• Metabolizable energy content of the feed.
: *cp_conc*	double	kg/kg	• Crude protein content of the feed.
: *prot_dg*	double	kg/kg	• Protein degradability of the feed.
: *p_conc*	double	kg/kg	• Phosphorus content of the feed.
: *s_conc*	double	kg/kg	• Sulphur content of the feed.
: *ee_conc*	double	kg/kg	• Ether-extractable content of the feed.
: *adip2cp*	double	kg/kg	• Proportion of crude protein that is insoluble in acid detergent.
: *ash_alk*	double	mol/kg	• Ash alkalinity of the feed.
: *max_passage*	double	kg/kg	• Maximum proportion of the feed that will pass undamaged through the gut of ruminants.

In addition, a property is registered for every supplement store defined in the component instance with the name of that store. The property is a structure:

Field	Type	Units	Description
: *stored*	double	kg	• Current amount of supplement in the store (fresh weight basis).
: *roughage*	Boolean		• TRUE i.f.f. the feed is a roughage.
: *dm_content*	double	kg/kg	• Dry matter content of the feed.
: *dmd*	double	-	• Dry matter digestibility of the feed (not including any portion that passes the gut undamaged).
: *me_content*	double	MJ/kg	• Metabolizable energy content of the feed.
: *cp_conc*	double	kg/kg	• Crude protein content of the feed.
: *prot_dg*	double	kg/kg	• Protein degradability of the feed.
: *p_conc*	double	kg/kg	• Phosphorus content of the feed.
: *s_conc*	double	kg/kg	• Sulphur content of the feed.
: *ee_conc*	double	kg/kg	• Ether-extractable content of the feed.
: *adip2cp*	double	kg/kg	• Proportion of crude protein that is insoluble in acid detergent.
: *ash_alk*	double	mol/kg	• Ash alkalinity of the feed.
: *max_passage*	double	kg/kg	• Maximum proportion of the feed that will pass undamaged through the gut of ruminants.

Configuration Details

Title: Supplement Component Description
Created by: A.D. Moore
Modified by: A.D. Moore
Processor: Microsoft Word 2002
Printed: 27 Apr 2006

Revision History

Version	Date	Changes
0.1	26 Mar 1998	First draft
0.2	31 Mar 1998	Revised after writing logic class
0.3	17 Aug 2000	ADIP:CP, P and S content added to initialisations
0.4	15 Dec 2003	Brought up-to-date for new protocol implementation. Ether-extractable content, ash alkalinity and max. passage added to attributes of supplements.
1.0	26 Oct 2005	Revisions to complete the documentation

Document Distribution Policy

All versions: Internal use, distribution with licensed copies of the SUPPLEMENT component