GREENHOUSE AUTOMATION
Specification Document

Module: 307CR Interactive Pervasive Computing
Coventry University, UK

Lecturers: Dr. Elena Gaura, Dr. James Brusey

Team: Greenhouse Gang

Members: Mandave Dosanjh (dosanjhm@coventry.ac.uk)
Daniel Knott (knottd@coventry.ac.uk)
Michael Straughan (straughm@coventry.ac.uk)
Péter Török (torokp@coventry.ac.uk)

Project website: http://greenhouse.progterv.info

Table of Contens

1 The Problem

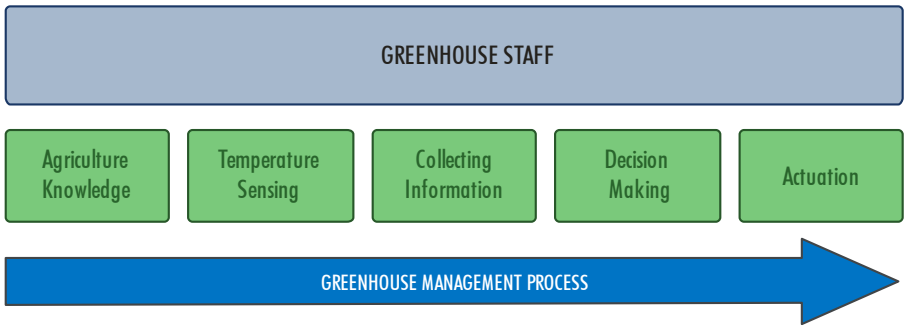
Greenhouse is built for efficiency. The aim of a greenhouse if to override the natural rules and make plant growing faster and more free from environmental effects, like temperature, amount of rain, humidity, etc.

A typical greenhouse is managed by horticulturists who have the knowledge about the plants needs. The efficiency of the greenhouse (which can be measured on the growth plants per crop land basis) is highly related to the decision made by the staff. The decision preparing needs background information to be better, as well as, other areas in life.

2 The Current System

The current running of a greenhouse is highly depends on the stuff. From the greenhouse planning till the last drop of water is managed by the stuff. However, time based watering systems are wide spread today.

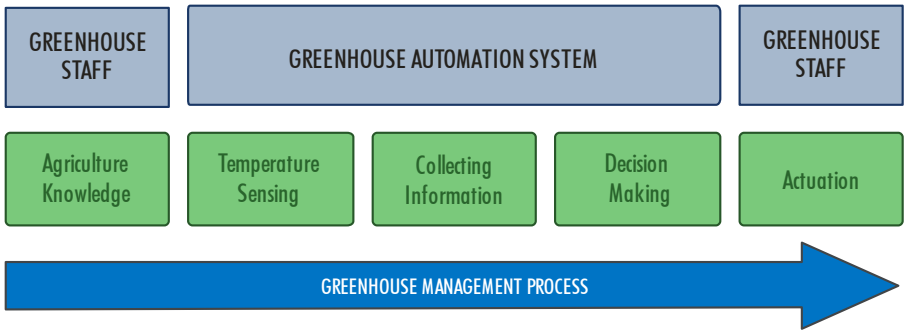
The System Layout Before The Project



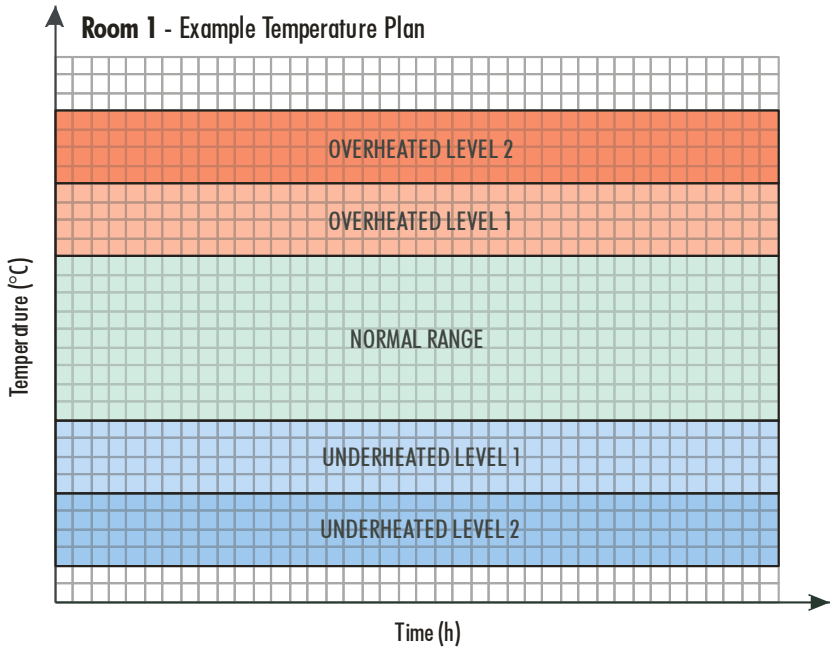
3 The Planned System

Our vision is to design a system which will measure and control both the air and soil temperature within a greenhouse as well as an automated sprinkler system. The agricultural knowledge still comes from the staff who are responsible for the right planning. But the periodic tasks, such as measuring the different values, then convert the data into information, then make an actual decision about the necessary steps can be done by a computer system.

The Planned System



For each room of the greenhouse the horticulturists have to plan the *rules*. The rules are responsible to trigger an event is the measured value in the range of the rule. An example setup shown on the graph. Each range on the graph is a rule. If the sensed value steps into one of the ranges, the action which is associated to that rule will be activated, and when the values leaves the range it will be turned off.



Actions are actuator device dependent; the commands will be stored to each action in the database and will be sent to the actuator device. In the current project the actuation will be only implemented as text alerts for the stuff, because the limits of project. For each mode of the actuator device needs a separate rule/action pair. For example *Overheated Level 1* sends the command to the heater device "`SET HEATING LEVEL 1`", and *Overheated Level 2* "`SET HEATING LEVEL 2`".

4 Deployed Hardware

4.1 Temperature sensing

- 4 Gumstix nodes with Bluetooth connections
- 20 temperature sensors (5 per gumstix, some embedded into soil, some in the air)
- Waterproof cases to protect sensors embedded into soil

4.2 Decision making and data storing

- Base station: PC or PDA with Bluetooth connections

5 "Undeployed" hardware (beyond the scope of the project):

- Hot water radiator system
- Sprinkler system
- Window motors
- Light bulbs

6 Deployed Software:

6.1 Base station

6.1.1 Database

A database that contains a example rules for an example greenhouse. The database stores the information about the room, nodes and sensors as entities in the system, and stores each physical location of the sensors which will be used for temperature interpolation.

6.1.2 Regular rule checker

Automated process which check all of the rules against the measured values, also responsible for device fault detection, if some of the nodes or sensors are not sending data. Generating command for the actuator subsystem.

6.2 User interface

The user interface will be deployed in a separate piece of application, which connect remotely to the base station over Ethernet network.

The GUI shows up the current status of the greenhouse from the sensed values, their difference from the expected values, and generating temperature maps.

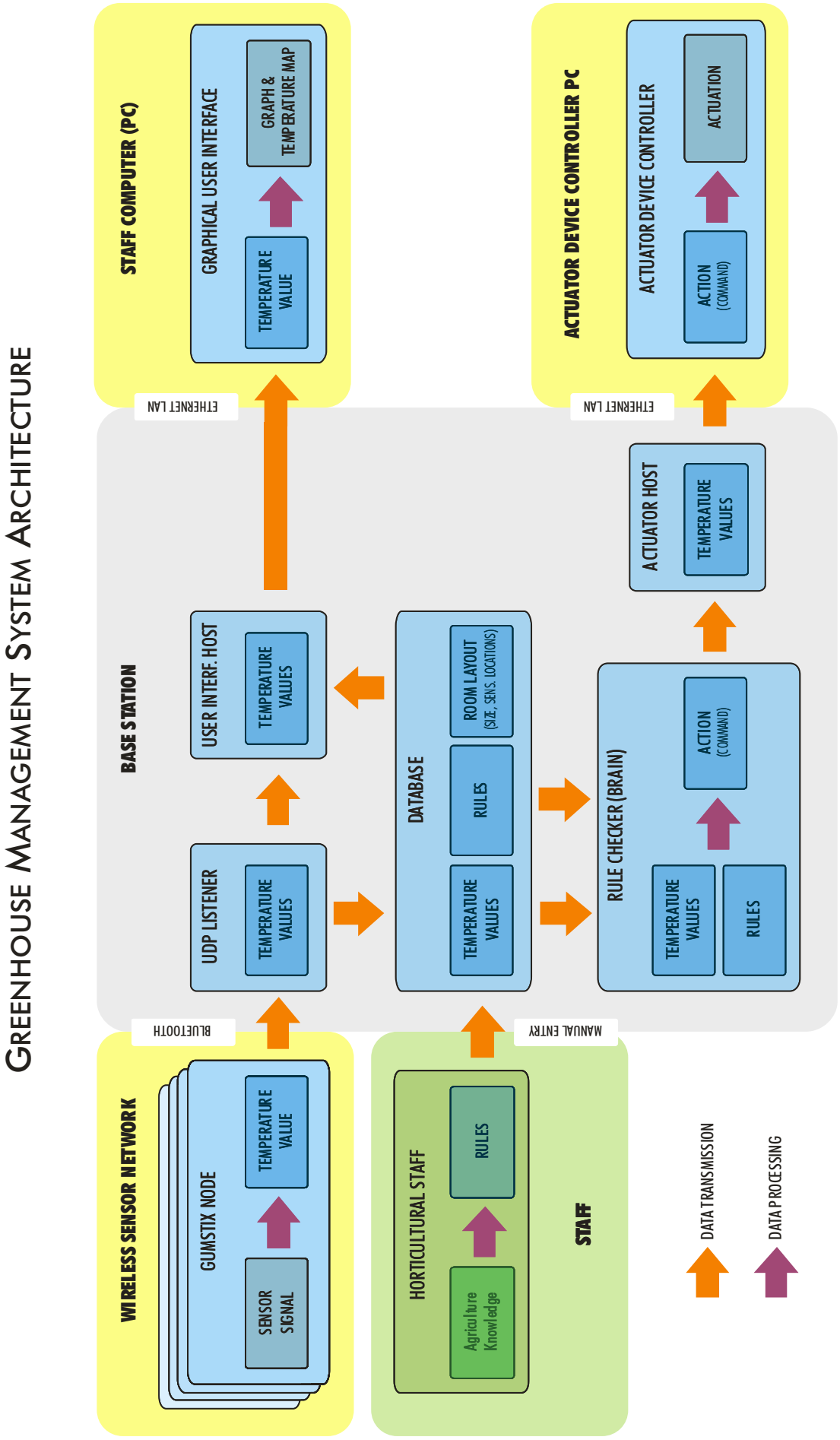
6.3 Actuator device controller

The actuator devices will be emulated with a mock up application which connects to the base station through the actuator interface (utilizing WCF-SOAP connection over network), but instead of actual device controlling it will shows up the commands as text.

6.4 Wireless Sensor Network

The sensor end of the system running python code on the gumstix, using Cogent Tools for sensor management and Bluetooth connection handling.

The communication will be UDP packet based, the packet losses won't be counted, but there is a time limit in which all the nodes should send data, if not, alert will be fired.


GREENHOUSE MANAGEMENT SYSTEM ARCHITECTURE
STAFF COMPUTER (PC)
GRAPHICAL USER INTERFACE
TEMPERATURE VALUE
GRAPH & TEMPERATURE MAP
ACTUATOR DEVICE CONTROLLER PC
ACTUATOR DEVICE CONTROLLER
ACTION (COMMAND)
ACTUATION
ETHERNET LAN
ETHERNET LAN
BASE STATION
USER INTERF. HOST
TEMPERATURE VALUES
ACTUATOR HOST
TEMPERATURE VALUES
UDP LISTENER
TEMPERATURE VALUES
DATABASE
ROOM LAYOUT (SIZE, SENS. LOCATIONS)
RULES
TEMPERATURE VALUES
RULE CHECKER (BRAIN)
ACTION (COMMAND)
TEMPERATURE VALUES
RULES
BLUETOOTH
MANUAL ENTRY
WIRELESS SENSOR NETWORK
GUMSTIX NODE
TEMPERATURE VALUE
SENSOR SIGNAL
HORTICULTURAL STAFF
RULES
Agriculture Knowledge
STAFF
DATA TRANSMISSION
DATA PROCESSING