

# Chapter 11: Managing InRoads Settings

## Chapter Overview

This chapter addresses the following major topics:

- InRoads Settings: the Big Picture
- End-user Expectations and Role
- Named Symbology
- Feature Styles
- Preferences

## **Section 1 - Managing InRoads: the Big Picture**

An engineer is obviously most productive when all of his effort is spent solving engineering problems and no time distracted by software overhead. A drafter is most productive when symbology standards are automatically adhered to when he does his drafting.

InRoads productivity is maximized when the InRoads is well managed. A well-managed InRoads environment allows users to focus on getting their work done without being distracted by software or even CAD standards issues. An interesting result of a well-managed InRoads environment is that it is easier to conform to standards than to deviate from them.

So what are specific characteristics of a well-managed InRoads environment?

- The end users can spend almost all of their time doing their work.
- Users spend no time looking up or adhering to graphics standards.
- Very little duplication of effort or rework occurs.
- Standards are adhered to automatically.
- Data is explicit and self-documenting.
- The users' needs are anticipated and satisfied.
- Unanticipated users needs are quickly and universally implemented into the environment.

The InRoads environment is best managed when a single person is responsible for facilitating InRoads productivity (many can participate, but ultimately one person should be responsible). This person should be very familiar with InRoads. An IT guy is not qualified, a CAD guru who is not familiar with InRoads is not qualified. The bigger the organization, the more important this person is to overall InRoads productivity.

The overall goal of this InRoads administrator is to understand, anticipate, implement and manage the needs of the entire InRoads team. He can save tremendous amounts of non-productive time and make QC automatic.

The primary deliverable of the InRoads Administrator is the preference file: building it, managing it, updating it, distributing it.

The end user's primary contribution to the InRoads Administrator is to provide feedback to the InRoads Administrator about issues related to quality control, ease-of-use, and design standards in InRoads. Typically if an end user is having a problem or running into a deficiency in the preference file, other users are too. If the InRoads Administrator can fix the issue, he can prevent all other users from wasting their time with the issue.

## **InRoads Management Technologies**

The primary InRoads management technologies are:

Named Symbologies  
Preferences  
Feature Styles

They will be discussed later in detail but in brief they can be defined as:

Named Symbologies: User-definable Named CAD settings presets.

Preferences: User-definable Named command-specific settings presets.

Feature Styles: User-definable “real world object” representations.

## **Example of an Attempt to Manage an InRoads Environment Well**

The preference file and cell library for this training manual is an attempt to create a well-managed InRoads environment. I, the author of this manual, tried to take every opportunity to take full advantage of the InRoads management technologies. I created a broad set of Feature Styles, I set up default and alternative settings (Preference Sets) for commands. I created a broad base of Named Symbologies and incorporated them into Preference Sets as often as I could. My goal was to have a manageable environment that provided reasonable-looking graphics for all the commands automatically. I saved tremendous amounts of time doing it by myself, ONCE, rather than having everyone taking this course each doing it themselves. If we were working as a team, I would want to know where the environment needs improvement; I would fix it and incorporate it into the standard dataset.

If I have been relatively successful you are doing this at the end of the class. If I have done a poor job you had to skip ahead and set up something I did not anticipate.

## **The end user’s expectation and role in Managing the InRoads Environment**

### **Preferences**

Preferences are definable, named, command-specific settings presets. They store (almost) all the settings in a command form. “Loading a Preference” restores all the form’s settings to previously saved values.

In a well-managed environment the default preferences should conform should be useful to productivity and conform to standards. If so, the end user get what he needs to perform his job in a way that conforms to standards **WITHOUT CHANGING ANY SETTINGS**. If the default Preference does not for some reason produce appropriate results, the end user should expect that a well-name Preference exists that will.

### Settings

If the end user is spending much time changing settings, he is wasting time. The less time the end users spend changing settings, the more productive they are. If at any time a command requires changing a large number of settings repeatedly, the user should 1) create a new Preference with those settings and 2) notify the InRoads Administrator of the need for a new preference.

### **Feature Styles**

Feature Styles are Object Types. When we define a Feature Style in InRoads, InRoads can then “understand” words and objects that we humans use. We should have a Feature Style for every object type that we want to track in InRoads.

For example, let us assume that our new boss has a fresh MBA and wants us to track “widgets” in our project. We can create a Feature Style in InRoads so that whenever an object is, in fact, a widget, we can tell InRoads through a variety of commands that the object is a widget. Later in the workflow, as we are using evaluation tools, InRoads can tell us when we’ve come across a widget.

The InRoads Administrator should have defined a broad array of Feature Style. The end user should only have to create new Feature Styles if something unusual, new or unanticipated needs to be tracked for a project. The InRoads Administrator should be notified unless the new Feature Styles will be unique to the project.

**Named Symbology**

Named Symbologies are a list of definable, named, CAD settings presets. Unlike Preferences they are not uniquely connected to a form. Any often-repeated symbology setting should be defined as a Named Symbology (as time permits). Named Symbologies can and should be used in command Preferences.

Typically, governing agencies have CAD graphic symbology requirements. An existing sewer pipe may have to be displayed in green on Level “underground\_sewer” (or 22 in MicroStation/J), with a weight of 3, with a dashed linestyle. A Named Symbology called “existing sewer” can be created with those settings. Once defined, users can select “existing sewer” when specifying the symbology for an existing sewer pipe.

Named Symbologies should be created when repeatedly specifying or changing symbology settings. In a well-managed environment, end users should seldom have to repeatedly change the same settings. A preference should have been created after a few iterations. Named Symbologies are important when it directly corresponds to a physical object type (Feature Style!) or pervades the software (“screened line” or “Title Text”).

**Section 2 - Managing InRoads: the Specifics**

**Named Symbology**

Named Symbology is managed from the Symbology Manager.

**1. Select InRoads>Tools>Symbology Manger**

The Symbology Manager form is invoked. It alphabetically lists all the Named Symbologies in the preference file.



**2. Scroll through the list to get an idea of the range of predefined symbology definitions.**

**3. Hit the New button.**

This may look intimidating, but it is really that complex. Everything InRoads displays can be segregated into three types of elements: Linear elements, Points, and Text. InRoads, like Civil Engineers, display three Views of objects: Plan/3D, Profile, and Cross Section.

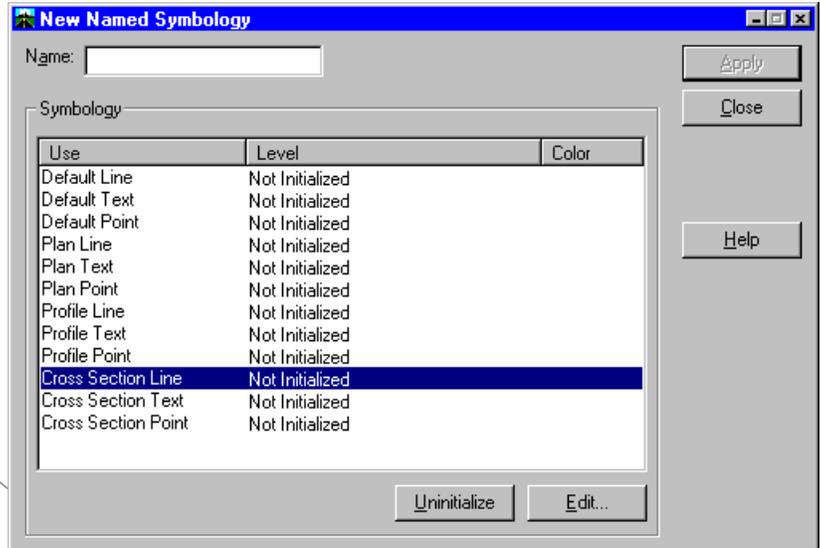
Three types multiplied by three views equals nine permutations. In most cases, an object is either only displayed in one View or the symbology requirements are identical in all views. In these cases the settings only need to be defined for Default Line, Default Point and/or Default Text.

The New Named Symbology form is invoked.

We will create a New Named Symbology for pavement striping.

4. Key in "Striping" in the Name field.

5.



### Line Symbology

6. In the Symbology list, double-click on the Default Line object.

These are the settings available for Lines.

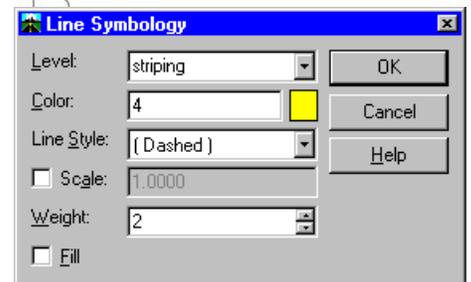
7. Key in "striping" for Level (if you are using MS/J or earlier, pick a whole number between 1 and 63).

8. Pick 4 (yellow) for Color.

9. Pick ( Dashed ) or 3 for Line Style.

10. Pick a Weight of 2

11. Hit OK.



Since we seldom (never) need to Annotate Striping (using InRoads commands, anyway) there is no need to define Text Symbology. Since we never need to show Striping as a Point there is no need to define Point Symbology.

With a single definition we have completely specified how InRoads should display striping.

12. Hit Apply

13. Do not close the form..

