

Early systems development often took place in a rather chaotic and haphazard manner, relying entirely on the skills and experience of the individual staff members performing the work. Today, many organizations still practice **Ad-hoc Development** either entirely or for a certain subset of their development (e.g. small projects).

The Software Engineering Institute at Carnegie Mellon University<sup>(2)</sup> points out that with **Ad-hoc Process Models**, "process capability is unpredictable because the software process is constantly changed or modified as the work progresses. Schedules, budgets, functionality, and product quality are generally (inconsistent). Performance depends on the capabilities of individuals and varies with their innate skills, knowledge, and motivations. There are few stable software processes in evidence, and performance can be predicted only by individual rather than organizational capability." <sup>(3)</sup>

### Figure 1. Ad-hoc Development

"Even in undisciplined organizations, however, some individual software projects produce excellent results. When such projects succeed, it is generally through the heroic efforts of a dedicated team, rather than through repeating the proven methods of an organization with a mature software process. In the absence of an organization-wide software process, repeating results depends entirely on having the same individuals available for the next project. Success that rests solely on the availability of specific individuals provides no basis for long-term productivity and quality improvement throughout an organization." <sup>(4)</sup>

---

<sup>(2)</sup> Information on the Software Engineering Institute can be found at <http://www.sei.cmu.edu>.

<sup>(3)</sup> Mark C. Paulk, Charles V. Weber, Suzanne M. Garcia, Mary Beth Chrissis, and Marilyn W. Bush, "Key Practices of the Capability Maturity Model, Version 1.1," Software Engineering Institute, February 1993, p 1.

<sup>(4)</sup> Mark C. Paulk, Bill Curtis, Mary Beth Chrissis, and Charles V. Weber, "Capability Maturity Model for Software, Version 1.1," Software Engineering Institute, February 1993, p 18.