"moderate" tie-in. Arguments of this type have been successful in recent antitrust cases.35

Finally, there is the argument, already suggested, that the defendant should be allowed to engage in the challenged conduct so that his incentive to innovate will not be chilled. He should not be forced to share the fruits of his innovative activity with "free riders" such as the plaintiffs. This argument has considerable emotional force and may constitute the court's unstated (but real) reason for overturning the jury's verdict in Data General.36 But previously discussed factors may tend to prevent its overt and explicit acceptance;37 First, the argument is more persuasive in militating against the imposition of new, affirmative duties on the defendant/inventor than it is in relieving him from existing prohibitions imposed by the law. Second, patent rights to inventions are invested by the law with the sanctity of "property rights," as are other statutorily recognized forms of intellectual property such as copyrights. But the proprietary rights involved in software are often not thus sanctified. It is questionable whether the courts can be persuaded to overtly accord software proprietors the same latitude in licensing that they now accord patent owners. Legislation explicitly recognizing proprietary rights in software is probably needed first. Yet, even though the courts may be unwilling to explicitly state that protection of proprietary rights justifies a tie-in, they may nonetheless decide in favor of the tie-in on those grounds, without openly admitting it.

One approach to justifying tie-ins argues that they are reasonable because they actually further the competitive process.


36 In the Data General case, Data General asserted at an early stage that its tie-in was a legitimate and necessary means of exploiting RDOS, but the court dismissed the claim out of hand (Data General 1, 490 F. Supp. at 1121). Data General argued that customers expect software to be supplied at little or no cost, that investment costs in software development can be recouped only by selling CPUs, and that the plaintiffs were would-be free riders on Data General's proprietary software development. The court replied that Data General should restructure its prices, if necessary, and reject the defense.

37 See text associated with Notes 29 and 30 above.

Richard H. Stern is a partner in the Washington, DC, law firm of Stern & Roberts. He specializes in computer and software law, patents, copyrights, trade secrets, and antitrust. Before he entered private law practice, he was Chief of the Intellectual Property Section and Patent Section in the US Justice Department's Antitrust Division. He has also been a trial attorney with the Federal Trade Commission and other federal agencies, a law clerk to Mr. Justice Byron White of the United States Supreme Court, and a law professor. Stern received a BA from Columbia College in 1953, a BSEE from Columbia Engineering School in 1954, and an LLB from Yale Law School in 1959. He is a member of Eta Kappa Nu. The author's address is Stern & Roberts, 2555 M Street NW, Washington, DC 20037.

The Data General II decision is far from the last word on the matter of hardware/software tie-ins. Its analysis of doctrine is probably unsound, but its implicit or covert message may speak more loudly than that analysis. The message seems to be that, one way or another, software proprietors will succeed in beating off efforts to make them share their products with competitors, even when the tools used to fence out the competitors are highly restrictive agreements.